Optimal synthesis into fixed XX interactions

Eric C. Peterson[1], Lev S. Bishop[2], and Ali Javadi-Abhari[2]

[1]IBM Quantum, San Jose, CA, USA
[2]IBM Quantum, Yorktown Heights, NY, USA

We describe an optimal procedure, as well as its efficient software implementation, for exact and approximate synthesis of two-qubit unitary operations into any prescribed discrete family of XX–type interactions and local gates. This arises from the analysis and manipulation of certain polyhedral subsets of the space of canonical gates. Using this, we analyze which small sets of XX–type interactions cause the greatest improvement in expected infidelity under experimentally-motivated error models. For the exact circuit synthesis of Haar-randomly selected two-qubit operations, we find an improvement in estimated infidelity by ≈31.4% when including $CX^{1/2}$ and $CX^{1/3}$ alongside the standard gate CX, near to the optimal limit of ≈36.9% obtained by including all fractional applications CX^α, $\alpha \in [0, 1]$.

1 Introduction

In this paper, we describe an optimal synthesis routine for two-qubit unitary operations which targets any discrete family of two-qubit gates, each locally equivalent to some $\exp(-iaXX)$. We refer to such gates as being of *XX–type* and note that this class includes all controlled unitaries. Gate sets of XX–type are common on contemporary platforms: the gate CX is an example, and synthesis routines for it have long been known to give rise to algorithmic schemes for universal quantum computation, making it an attractive target for device engineers. The physical processes which give rise to the operation CX can typically be truncated to produce "fractional applications" CX^α for $0 \leq \alpha \leq 1$, each of which is *also* of XX–type, giving rise to an infinite family of further examples.[1] Such fractional applications can be found on devices based on superconducting qubits (e.g., IBM's), as well as on those based on ion traps. Though not required for universal computation, the availability of these "overcomplete" basis gates has the potential to yield more efficient synthesized circuits, particularly if the error magnitude of CX^α correlates with α: while the universally programmable CX circuit invokes CX three times, the universally programmable fractional CX circuit invokes CX^α, CX^β, and CX^γ with $\langle \alpha + \beta + \gamma \rangle = 3/2$.[2]

In practice, however, these parametric families are difficult to operate. The relationship between the degree of physical process truncation and the value α is often nonlinear and prone to imperfect measurement, and constraints in the steering electronics (e.g., waveform sample rate) can make truncations unavailable below some threshold, so that wholesale use of these families may be prohibited on realistic hardware. Still, precision can be guaranteed for any particular value of α, which gives rise to the following question:

Question. Given a fixed "calibration budget" which permits the tuning of n fractional operations, which set of values $\alpha_1, \ldots, \alpha_n$ maximizes (average-case) device performance? How does one efficiently find expressions for generic two-qubit unitaries in terms of these operations? How does one simultaneously guarantee the optimality of such expressions, as measured against device performance?

We answer this question fully. Our fundamental results are an efficient test for when a two-qubit unitary operation admits expression as a circuit using any particular sequence of XX–interaction strengths $(\alpha_1, \ldots, \alpha_n)$ with local gates interleaved (Theorem 4.1), an efficicent synthesis routine for manufacturing such circuits (Procedure 6.1, Theorem 5.5), and an efficient routine for producing the best approximation (in average gate fidelity) within the set of such circuits (Procedure 6.8).[3] These tools combine to give an optimal synthesis scheme for reasonably behaved cost functions (e.g., average gate infidelity). An implementation of our technique can be found in Qiskit's `quantum_info` subpackage as the class `XXDecomposer` [3], where it can be verified that it outperforms blind numerical search in both wall time and output quality (cf. Figure 8). We leverage these results to explore the design space of gate set extensions where we constrain $\alpha_1, \ldots, \alpha_n$ to be drawn from

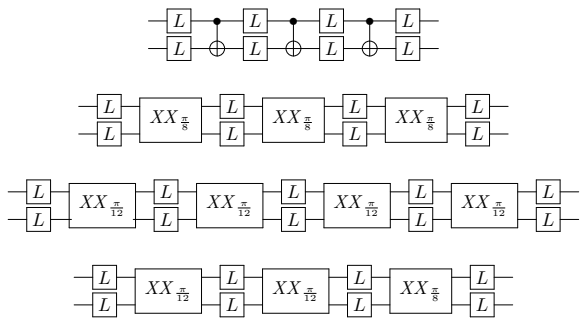
Eric C. Peterson: peterson.eric.c@gmail.com
Lev S. Bishop: lsbishop@us.ibm.com
Ali Javadi-Abhari: Ali.Javadi@ibm.com

[1]Up to local equivalence, this family is an exhaustive set of examples.

[2]See [1] for numerical justification or [2] for a proof.

[3]To our knowledge, this is the first optimal synthesis algorithm which targets a "heterogeneous" set of multiqubit interactions.

$$U = \begin{pmatrix} 0.098+0.121j & -0.26-0.391j & -0.443-0.322j & -0.662+0.133j \\ 0.748-0.132j & 0.372-0.084j & -0.432-0.027j & 0.261-0.148j \\ -0.137-0.365j & -0.453+0.334j & -0.28-0.547j & 0.296-0.257j \\ -0.474-0.147j & 0.44-0.352j & -0.265-0.255j & 0.328+0.439j \end{pmatrix}$$



	Relative infidelity		
Gateset description	[5]	[6]	Ours
$\{XX_{\frac{\pi}{4}}\}$ (baseline)	133%	**100%**	**100%**
$\{XX_{\frac{\pi}{8}}\}$	67%	67%	**50%**
$\{XX_{\frac{\pi}{12}}\}$	56%	56%	**44%**
$\{XX_{\frac{\pi}{4}}, XX_{\frac{\pi}{8}}, XX_{\frac{\pi}{12}}\}$	n/a	n/a	**39%**

Figure 1: Four syntheses of a two-qubit operator $U \in SU(4)$ with canonical coordinate $(0.968, 0.273, 0.038)$. **(1)** An exact, optimal synthesis into a triple of CX gates. **(2)** An exact, optimal synthesis into a triple of $XX_{\frac{\pi}{8}}$ gates. **(3)** An exact, optimal synthesis into four $XX_{\frac{\pi}{12}}$ gates. **(4)** An exact, optimal synthesis into a mixed set of gates. Finally, we include the relative infidelity costs of these syntheses in an error model where XX gate infidelity is linearly related to the parameter with a small affine offset. **Relative infidelity:** We assume that XX gate infidelity is linearly related to the parameter with a small affine offset and that infidelity is approximately additive over that of its gates. We refer to the infidelity of circuit (1) as the "baseline", and we report as "relative infidelity" the percentage of this value achieved by other synthesis strategies. *See Figure 8 and Section 7 for similar statistics where U is allowed to range.*

	Gates emitted		
n	CX	$\{XX_{\frac{\pi}{4}}, XX_{\frac{\pi}{8}}, XX_{\frac{\pi}{12}}\}$	Rel. Infidelity
3	CX^6	$XX^2_{\pi/12} \cdot XX^2_{\pi/8}$	27.8%
4	CX^{12}	$XX^6_{\pi/12} \cdot XX^3_{\pi/8}$	29.2%
5	CX^{20}	$XX^{12}_{\pi/12} \cdot XX^4_{\pi/8}$	30.0%
6	CX^{30}	$XX^{20}_{\pi/12} \cdot XX^5_{\pi/8}$	30.6%
7	CX^{42}	$XX^{30}_{\pi/12} \cdot XX^6_{\pi/8}$	31.0%

Figure 2: Qiskit syntheses of QFT circuits over n qubits, targeting a family of qubits supporting either $\mathcal{S} = \{CX\}$ or $\mathcal{S} = \{XX_{\frac{\pi}{4}}, XX_{\frac{\pi}{8}}, XX_{\frac{\pi}{12}}\}$ and with all-to-all connectivity. At right, we include the expected circuit infidelity, reported as a fraction of that of traditional synthesis methods, under the assumptions that XX gate infidelity is linearly related to the parameter with a small affine offset and that infidelity is approximately additive. In the limit of a large qubit count, the expected infidelity of a QFT circuit synthesized to the fractional gate set drops by two-thirds that of the standard CX–based gate set. We don't intend circuit synthesis for QFTs to be a "killer app", but rather as evidence that these methods are not limited to random inputs.

some small, fixed set of pretuned angles. For experimentally realistic error models,[4] our main findings are first that including first $CX^{1/2}$ and then $CX^{1/3}$ give significant improvement over CX alone at several common tasks[5], and second that these two gates capture almost all of the benefit of allowing $\alpha_1, \ldots, \alpha_n$ to be drawn without constraint (cf. Figure 15). Finally, we note that some of our proofs rely on using a computer algebra system to manipulate polytopes, and we have released both this framework and the proof software under the Qiskit umbrella of packages as `monodromy` [4].

Related literature

We give a non-exhaustive survey of existing ideas.

[7] Shende, Markov, and Bullock showed how to synthesize optimal-depth circuits for CX–based gate sets.

[8] Cross et al. extended that to synthesize optimal-infidelity circuits for CX–based gate sets.

[5, 9, 10] Zhang, Vala, Sastry, and Whaley (see also Zhang, Ye, and Guo) gave a synthesis method for gate sets based on controlled unitary operations. Our method constructs circuits out of the same building blocks, but our method for how to arrange those blocks is different. This difference in synthesis strategy gives us *optimal* circuits, whereas often[6] they miss by a constant offset.

[11] Peterson et al. showed how to detect when a two-qubit unitary operation admits expression using a given circuit type with sufficiently many freely ranging local operations. This gives rise to a method for analyzing the optimality of a synthesis strategy, but it does not show how to perform the actual synthesis.

[12] Earnest, Tornow, and Egger have shown how to produce the entire family of XX–type interactions from a particular pulse-level implementation of CX, which then permits the use

[4] In practice, we find that these models amount to a *weighted* count of circuit elements, where the weights depend linearly on the exponents α_j.

[5] Specifically, we examine synthesis of random unitaries, as encountered during whole-circuit resynthesis, and reported on in Figure 18; and we exaxmine synthesis of certain structured operators like QFT, as encountered in peephole optimization of highly structured circuits, and reported on in Figure 2.

[6] More exactly, for $\frac{1}{2}(5 + 4\log(2) - 6\log(3)) \approx 59\%$ of possible native XX interactions; cf. [5, Figure 2].

of a straightforward synthesis method. Their method does not extend to other hardware implementations of CX, including the implementation used by the IBM group to achieve quantum volume 64 [13], which does not lend itself to non-calibrated scaling.

[14] Huang et al. have recently shown how to synthesize (optimal) circuits for a gateset containing $\sqrt{ISWAP}$, a particular "fractional $ISWAP$". Despite surface similarities, our results depart substantially: $\sqrt{ISWAP}$ is not of XX–type; they work with the fixed gate $\sqrt{ISWAP}$ as opposed to an unknown family of fractional applications; and they consider circuits of depth at most 3.

[6] Lao et al. have used brute-force numerical search to uncover which fractional interactions might be valuable to include in a native gate set put to various specific purposes. This has substantial overlap with our discussion of gate set optimization, but it does not solve the synthesis problem: numerical search is both non-optimal and two orders of magnitude slower than our direct synthesis.

The case of one qubit

To give a sense of our methods and results, let us analyze the analogous problem for one-qubit unitaries: decomposition into a fixed set of fractional X–rotations and unconstrained Z–rotations. The fixed X–rotation most typically available is $X_{\frac{\pi}{2}}$, and an $X_{\frac{\pi}{2}}$–based circuit can be synthesized for a unitary U through "Euler ZYZ decomposition". Namely, there are angle values θ, ϕ, and λ which satisfy

$$U = Z_\phi \cdot Y_\theta \cdot Z_\lambda = Z_{\phi+\pi} \cdot X_{\frac{\pi}{2}} \cdot Z_{\theta+\pi} \cdot X_{\frac{\pi}{2}} \cdot Z_\lambda,$$

easily calculable by diagonalizing UU^T. Since U can freely range, the right-hand side of this equation gives a universally programmable quantum circuit. A downside to this circuit is that the operational cost of U is always that of a pair of $X_{\frac{\pi}{2}}$ gates, even if U itself is a "small" rotation of the Bloch sphere.[7]

For circuits based on other choices of choices of X–rotation angles, such as

$$Y_\theta = Z_{\iota'} \cdot X_\psi \cdot Z_\iota \cdot X_{\psi'} \cdot Z_{\iota''},$$

one perform some mathematical analysis to discern the limited set of synthesizable operations Y_θ, ultimately arriving at the critical relationship

$$\cos\theta = \cos\psi \cdot \cos\psi' - \cos\iota \cdot \sin\psi \cdot \sin\psi',$$

and the remaining parameters ι', ι'' can be explicitly determined by inspecting complex phases. Varying ι, ι', and ι'', this trigonometric equation admits a solution precisely when θ satisfies

$$|\psi - \psi'| \leq \theta \leq \pi - |\pi - (\psi + \psi')|\,.$$

Let us refer to this interval as $I_{\psi,\psi'}$. In the same manner, a longer sequence of interactions $X_{\psi_1}, \ldots, X_{\psi_n}$ interleaved with Z–rotations gives rise to a corresponding interval $I_{\psi_1,\ldots,\psi_n}$ of achievable values of θ.

Suppose that any given gate X_ψ, with $\psi \in [0, \pi]$, can be made available in an experimental setting with infidelity

$$m \cdot \left(\frac{\pi}{2}\right)^{-1} \cdot \psi + b$$

for some error model parameters m and b, and at a fixed calibration cost per gate. We seek a small set of gates $\{X_{\psi_1}, \ldots, X_{\psi_n}\}$ so that the intervals constructed above cover the possible values of θ so as to minimize the expected infidelity cost of a given operation. For instance, Figure 3 shows the relevant intervals for the gate set $\{X_{\frac{\pi}{2}}, X_{\frac{\pi}{3}}\}$. Several aspects of this goal can also be understood with additional nuance:

"Expected": The distribution of operations U to be synthesized will affect the relative importance of the various choices of ψ. A safe assumption is that U is drawn according to the Haar distribution, in which case the distribution of angle values θ is given by $p(\theta)d\theta = \frac{1}{2}\sin(\theta)d\theta$.

"Cost": In addition to the operational cost of a synthesized circuit (i.e., the cost from gate applications), one can also incorporate a cost stemming from synthesizing some θ' rather than the requested θ. There are then some circumstances where it is profitable to *deliberately* mis-synthesize Y_θ as $Y_{\theta'}$, provided the difference between θ and θ' is small and the difference in operational cost between the two circuits is large. Average gate infidelity gives a popular embodiment of this idea, where the fidelity of two one-qubit operations is given by the formula

$$F_{\text{avg}}(Y_\theta, Y_{\theta'}) = \frac{2 + \cos(\theta - \theta')}{3}.$$

Against this yardstick, the $\theta' \in I_{\psi,\psi'}$ which gives the best approximation to a $\theta \notin I_{\psi,\psi'}$ occurs at one of the interval endpoints.

"Given operation": Rather than synthesizing the operation U requested, the compiler can choose to inject a reversible logic operation R and its inverse R^{-1} into the program, synthesizing the composite $U \cdot R$ and either commuting R^{-1} forward through the circuit or absorbing its effect into software. This option can be used to further shape the expected distribution of inputs. For single-qubit operations, a typical choice of R is

[7] One can set aside special cases when θ is zero or $\frac{\pi}{2}$, but these are probability-zero events in common measures.

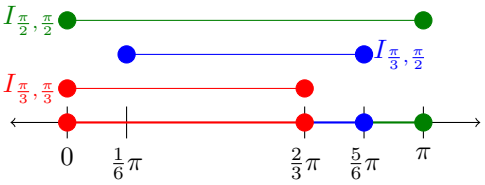


Figure 3: The optimal synthesis regions for the one-qubit gate set $\{X_{\frac{\pi}{2}}, X_{\frac{\pi}{3}}, Z_{cts}\}$. The interval being covered is the set of angles $[0, \pi]$ appearing as the middle parameter in a ZYZ–decomposition of a generic $U \in PU(2)$.

the classical logic gate X_π, which has the effect of trading X_θ for $X_{\pi-\theta}$.[8]

Considering only exact synthesis for now, we compute the following expected (i.e., Haar-averaged) average gate infidelities for various gate sets:

$X_{\frac{\pi}{2}}$**:** $2m + 2b$, the standard decomposition, used as a baseline.

$X_{\frac{\pi}{3}}$**:** $\frac{3}{2}m + \frac{9}{4}b$, an improvement of $\approx 25\%$ over the baseline, provided $b \ll m$.

$\{X_{\frac{\pi}{2}}, X_{\frac{\pi}{3}}\}$**:** $\frac{19-\sqrt{3}}{12}m + 2b$, an improvement of $\approx 28\%$ over the baseline, provided $b \ll m$.

$\{X_{cts}\}$**:** In the continuous limit with all gates X_θ available, the cost becomes $m + b$, an improvement of 50% over the baseline.

These values can be further improved by considering approximate synthesis, mirrored synthesis, or both.

Outline

Our analysis of the two-qubit case follows along the same lines as above, and in the same order.

Section 2: Generalizing Euler decomposition, we give a lightning review of KAK decomposition as specialized to two-qubit unitary operations.

Section 3: We describe a more detailed plan of attack on the two-qubit problem, outlining the steps in the proofs to come.

Section 4: Generalizing the interval $I_{\psi,\psi'}$, we give a compact description of which two-qubit gates are accessible to a circuit built out of a fixed sequence of XX–type interactions with one-qubit operations interleaved (Theorem 4.1). This leverages previous work of Peterson et al. [11]: it detects when a two-qubit operation admits synthesis as a circuit of a certain type, but it does not indicate how to produce the circuit.

[8]For two-qubit operations, a common choice of R is the classical logic gate $SWAP$.

Section 5: Generalizing the formula relating $\cos\theta$ and $\cos\iota$, we single out a method for choosing local circuit parameters which are simple to analyze (Theorem 5.3). We then compare with Section 4 and prove that each of these restricted circuits nonetheless exhaust the space of possibilities (Theorem 5.5).

Section 6: Generalizing the discussion around cost, we provide an efficient method to find the best approximation within a given circuit family (Theorem 6.10), and we couple it to the preceding results to produce the promised efficient synthesis method (Procedure 6.1, Procedure 6.8).

Section 7: Generalizing the calculations of expected cost for various gate sets, we use experimental data to justify the use of a particular error model as a cost function (Definition 7.3), study the effect of choice of gate set (Example 7.6), and describe what is left to gain in the large limit of fractional gate count (Remark 7.8).

We give a small example of the effectiveness of these techniques as applied to a random operator in Figure 1 and to a family of structured operators in Figure 2, reserving further analysis for Section 7.

Conventions

We use the following abbreviations throughout:

$$c_\theta = \cos(\theta), \qquad s_\theta = \sin(\theta), \qquad x_+ = \sum_j x_j.$$

2 Résumé on two-qubit unitaries and the monodromy map

We briefly recall the theory of Cartan decompositions as it applies to two-qubit unitary operations and its role in circuit synthesis.

Lemma 2.1 ([15], [16], [17], [11])**.** *Let CAN denote the following two-qubit gate:*

$$CAN(a_1, a_2, a_3) = \exp(-i(a_1 XX + a_2 YY + a_3 ZZ)) =$$

$$\begin{pmatrix} e^{ia_3}c_{a_1-a_2} & 0 & 0 & -ie^{ia_3}s_{a_1-a_2} \\ 0 & e^{-ia_3}c_{a_1+a_2} & -ie^{-ia_3}s_{a_1+a_2} & 0 \\ 0 & -ie^{-ia_3}s_{a_1+a_2} & e^{-ia_3}c_{a_1+a_2} & 0 \\ -ie^{ia_3}s_{a_1-a_2} & 0 & 0 & e^{ia_3}c_{a_1-a_2} \end{pmatrix}.$$

Any two-qubit unitary operation $U \in PU(4)$ can be written as

$$U = L \cdot CAN(a_1, a_2, a_3) \cdot L',$$

where $L, L' \in PU(2)^{\times 2}$ are local gates and a_1, a_2, a_3 are (underdetermined) real parameters. □

Definition 2.2 ("Canonical decomposition", cf. [11])**.** In Lemma 2.1, there is a unique triple (a_1, a_2, a_3) satisfying $a_1 \geq a_2 \geq a_3 \geq 0$, $\frac{\pi}{2} \geq a_1 + a_2$, and one of

either $a_3 > 0$ or $a_1 \leq \frac{\pi}{4}$. Such a triple is called a *positive canonical coordinate*, and we denote the space of such as $\mathfrak{A}_{C_2}$. This unicity determines a function $\Pi\colon PU(4) \to \mathfrak{A}_{C_2}$, called the *monodromy map*. Away from the plane $a_3 = 0$, this function is continuous.[9]

Example 2.3. Here are the positive canonical triples for some familiar gates:

$$\begin{aligned}\Pi(I) &= (0,0,0),\\ \Pi(SWAP) &= \left(\frac{\pi}{4},\frac{\pi}{4},\frac{\pi}{4}\right),\\ \Pi(CX) &= \left(\frac{\pi}{4},0,0\right).\end{aligned}$$

Generalizing the last example, the positive canonical triple for any controlled unitary gate has the form $(a_1, 0, 0)$; we say that such an operation is *of XX–type*, and we abbreviate such gates to

$$XX_{a_1} = CAN(a_1, 0, 0).$$

Specifically, the fractional gate CX^α is of XX–type, with coordinate $\Pi(CX^\alpha) = (\alpha \cdot \frac{\pi}{4}, 0, 0)$, so that

$$CX^\alpha = (H \otimes I) \cdot XX_{\alpha\cdot\frac{\pi}{4}} \cdot (H \cdot Z_{-\frac{\pi}{2}} \otimes Z_{-\frac{\pi}{2}}) \equiv XX_{\alpha\cdot\frac{\pi}{4}}.$$

From this perspective, the varying coordinate measures interaction duration or interaction strength, so that smaller values give rise to less entanglement.

For us this apparatus has two main uses, captured in the following pair of results:

Lemma 2.4. *A pair of two-qubit operations U and V are said to be locally equivalent when there exist local gates $L, L' \in PU(2)^{\times 2}$ with $U = L \cdot V \cdot L'$. This condition holds if and only if $\Pi(U) = \Pi(V)$.* □

Theorem 2.5 ([11]; [4])**.** *Let $\mathcal{S}, \mathcal{S}' \subseteq PU(4)$ be two sets of two-qubit operations whose images $\Pi(\mathcal{S}), \Pi(\mathcal{S}') \subseteq \mathfrak{A}_{C_2}$ through Π are polytopes (e.g., a set of isolated points). The image of the set*

$$\mathcal{S}\cdot PU(2)^{\times 2}\cdot\mathcal{S}' = \left\{ S \cdot (L \otimes L') \cdot S' \;\middle|\; \begin{array}{c} S \in \mathcal{S}, S' \in \mathcal{S}', \\ L, L' \in PU(2) \end{array} \right\}$$

through Π is then also a polytope. Given explicit descriptions of the input polytopes as families of linear inequalities, the output polytope can also be so described. □

Our work will lead us directly into considering families of two-qubit gates and their parametrizations, so we introduce some attendant language.

Definition 2.6. A *gate set* is any collection of two-qubit unitaries; typically we will consider gate sets which are made up of finitely many controlled unitaries. For a gate set $\mathcal{S}$, an $\mathcal{S}$–circuit is a finite sequence of members of $\mathcal{S}$ and local gates. The operation which it enacts is given by the product of the sequence elements. A *circuit shape*[10] is a circuit-valued function

$$C\colon \theta \mapsto (L_0(\theta), S_1, \ldots, L_{n-1}(\theta), S_n, L_n(\theta)),$$

where each L_j is a parametrized local operator and each $S_j \in \mathcal{S}$ is fixed.

It can be convenient to place further restrictions on the L_j (e.g., that they consist only of Z–rotations), but absent explicit mention we take each L_j to surject onto $PU(2)^{\times 2}$. In this surjective case, the sequence $(S_1, \ldots, S_n)$ determines the image of C, and it follows from Theorem 2.5 that the image of $\Pi \circ C$ in $\mathfrak{A}_{C_2}$ is a polytope, called the *circuit polytope* of C (or of $(S_1, \ldots, S_n)$). In the case that $\mathcal{S}$ consists of gates of XX–type, locally surjective circuits can be further identified with the underlying sequence of interaction strengths $(\alpha_1, \ldots, \alpha_n)$ with $S_j \equiv XX_{\alpha_j}$.

Remark 2.7. The coordinate system given in Definition 2.2 is not unique: a similar theorem holds for any choice of "Weyl alcove" in $\mathfrak{pu}_4$. When $\mathfrak{g}$ is the Lie algebra of a simply connected Lie group (e.g., $\mathfrak{su}_4$), each Weyl alcove is related to every other by a discrete set of linear transformations including reflections and shears. Without the simply-connected hypothesis (e.g., $\mathfrak{pu}_4$), they are related by linear transformations and "scissors congruence". Our choice of coordinate system differs from that used in previous work of Peteron et al. [11] by a nontrivial scissors congruence, effectively replacing the condition stated there,

$$(\operatorname{LogSpec} U)_3 + 1/2 > (\operatorname{LogSpec} U)_1,$$

by the alternative

$$(\operatorname{LogSpec} U)_2 + (\operatorname{LogSpec} U)_3 > 0.$$

Remark 2.8 (cf. Remark 4.5). Later, it will be convenient for us to consider a variant of positive canonical triples which are not required to be sorted. Unsorted triples (a_1, a_2, a_3) which become positive canonical triples upon sorting are those which satisfy $0 \leq a_j$ and $a_j + a_k \leq \frac{\pi}{2}$ for all choices of j and k. The set of such triples still gives a convex polytope.

3 XX-based synthesis: Strategy

Let us turn to the task of synthesizing for any two-qubit unitary operator U and gate set $\mathcal{S}$ consisting of XX interactions an $\mathcal{S}$–circuit C modeling U. To set the stage for our strategy, suppose instead that C is

[9] Near the plane $a_3 = 0$, the function Π becomes continuous after imposing the identification $(a_1, a_2, 0) \sim (\frac{\pi}{2} - a_1, a_2, 0)$.

[10] Also commonly called a *(circuit) template*.

given. We can produce from it a sequence of truncations C_j that retain steps 1 through j. Each C_j is also a circuit modeling some other unitary operator U_j, and if C is optimal for circuits modeling U against some well-behaved cost function (e.g., operation count), then each C_j will be so optimal for U_j. The images $p_j = \Pi(U_j) \in \mathfrak{A}_{C_2}$ of these intermediate operators then describe a path through the Weyl alcove, where the j^{th} step in the path belongs to the region P_j of operations whose optimal circuits take the form of C_j.

Since our goal is to construct C, we might instead begin by constructing the path $(p_j)_j$, subject to the two constraints:

1. p_j lies in P_j.
2. The hop from p_j to p_{j+1} is given by some nice circuit.

In order to understand the first constraint, we give a compact description of P_j by way of describing the circuit polytope associated to an arbitrary sequence of XX–type interactions. We call this the *global theorem* (Theorem 4.1) since it describes the large-scale structure of the problem and does not reference the individual point p_j. Though our main tool here is Theorem 2.5, for a generic sequence of interactions it can only guarantee an exponential-sized family of convex bodies, themselves each of increasing facet complexity. It is a special feature of interactions of XX–type that the associated circuit polytopes have a fixed number of convex bodies, each of fixed complexity, independent of the sequence length.

To understand the second constraint, we choose a particular "nice circuit" and analyze the effect under Π of appending such a circuit to a canonical gate (Lemma 5.1), resulting in a family of constraints we call "interference inequalities" (Theorem 5.3). This, too, is specific to our case: even for interactions of XX–type, not all choices of unit circuit have a discernable image under Π, nevermind a polytope.

We complete the program by linking these two together in the *local theorem* (Theorem 5.5): we show that for any $p_{j+1} \in P_{j+1}$, we can always find a $p_j \in P_j$ linked to p_{j+1} by one of these simple circuits. This argument can then be reorganized into a constructive, efficient synthesis routine (Procedure 6.1). Additionally, we show how to select a point $p' \in P_j$ which is the best approximation by average gate infidelity to $p = \Pi(U)$ (Theorem 6.10).

4 The global theorem

One of our overall goals is to describe the set of positive canonical triples whose optimal circuit implementation uses a sequence of interaction strengths $(\alpha_1, \ldots, \alpha_n)$. This can be accomplished by describing those positive canonical triples which admit *any* such circuit implementation, even if suboptimal. Optimality can then be enforced by taking a complement against positive canonical triples which admit superior circuit implementations. In this section, we accomplish this goal, summarized in the following Theorem:

Theorem 4.1. *Let $(\alpha_j \in [0, \frac{\pi}{4}])_j$ be a sequence of interaction strengths, and let (a_1, a_2, a_3) be a positive canonical coordinate. The canonical operator $CAN(a_1, a_2, a_3)$ admits a presentation as a circuit of the form*

$$L_0 \cdot XX_{\alpha_1} \cdot L_1 \cdot \cdots \cdot L_{n-1} \cdot XX_{\alpha_n} \cdot L_n,$$

where L_j are local operators, if and only if either of the following two families of linear inequalities is satisfied:

$$\begin{cases} \alpha_+ & \geq & a_1 + a_2 + a_3, \\ \min_k \alpha_+ - 2\alpha_k & \geq & -a_1 + a_2 + a_3, \\ \min_{k \neq \ell} \alpha_+ - \alpha_k - \alpha_\ell & \geq & a_3; \end{cases}$$

$$\begin{cases} -\frac{\pi}{2} + \alpha_+ & \geq & -a_1 + a_2 + a_3, \\ \frac{\pi}{2} + \min_k \alpha_+ - 2\alpha_k & \geq & a_1 + a_2 + a_3, \\ \min_{k \neq \ell} \alpha_+ - \alpha_k - \alpha_\ell & \geq & a_3. \end{cases}$$

We respectively refer to the first, second, and third inequalities in each family as the strength, slant, and frustrum bounds.

Important Remark. From a physical perspective, the circuit polytope ought to be invariant under injecting extra zero-strength interactions into the defining sequence of interaction strengths. Accordingly, we always treat expressions like "$\min_{k \neq \ell} \alpha_+ - \alpha_k - \alpha_\ell$" as if the sequence were padded by arbitrarily many zero entries.

Proof of Theorem 4.1. For the base case, note that the empty list of interaction strengths yields the polytope

$$a_1 = a_2 = a_3 = 0,$$

which agrees with the set of circuits locally equivalent to the identity interaction.

Suppose then that we have established the claim for a sequence of interaction strengths $(\alpha_1, \ldots, \alpha_n)$, and we would like to establish the claim for $(\alpha_1, \ldots, \alpha_n, \beta)$ for some new interaction strength β. By allowing the $(n+1)$ different strengths to range, we note the region in the claim is naturally expressed as a polytope in $(n+1)+3$ dimensions. In fact, we can reduce it to a certain 6–dimensional polytope as follows: writing α' and α'' respectively for the largest and second-largest elements in the hypothesized sequence of interaction strengths, we may rewrite the inequality fami-

lies above as

$$\begin{cases} \alpha_+ & \geq & a_1+a_2+a_3, \\ \alpha_+ - 2\alpha' & \geq & -a_1+a_2+a_3, \\ \alpha_+ - \alpha' - \alpha'' & \geq & a_3; \end{cases}$$
$$\begin{cases} -\frac{\pi}{2} + \alpha_+ & \geq & -a_1+a_2+a_3, \\ \frac{\pi}{2} + \alpha_+ - 2\alpha' & \geq & a_1+a_2+a_3, \\ \alpha_+ - \alpha' - \alpha'' & \geq & a_3. \end{cases}$$

with the additional constraints

$$n \cdot \frac{\pi}{4} \geq \alpha_+, \quad \alpha_+ \geq \alpha' + \alpha'', \quad \frac{\pi}{4} \geq \alpha' \geq \alpha'' \geq 0.$$

Altogether, these statements over $a_1, a_2, a_3, \alpha_+, \alpha', \alpha''$ describe a pair of convex polytopes in 6–dimensional space.

Theorem 2.5 gives an explicit, finite family of linear inequalities (the "monodromy polytope") so that $a_1, a_2, a_3, a_1', a_2', a_3', b_1, b_2, b_3$ satisfies the constraints if and only if there is a local gate L and a local equivalence

$$CAN(a_1,a_2,a_3)\cdot L \cdot CAN(a_1',a_2',a_3') \equiv CAN(b_1,b_2,b_3).$$

We combine this with the polytope from the inductive hypothesis so that its coordinates are shared with (a_1, a_2, a_3) and the coordinates (a_1', a_2', a_3') are set to $(\beta, 0, 0)$. This produces a union of convex polytopes in 10–dimensional space, a point of which simultaneously captures:

$(\alpha_+, \alpha', \alpha'')$**:** Values extracted from the prefix of interaction strengths.

(a_1, a_2, a_3)**:** A positive canonical coordinate which admits expression as an XX–circuit as in the inductive hypothesis.

β**:** A new interaction strength.

(b_1, b_2, b_3)**:** A canonical coordinate which admits expression as a concatenation of the aforementioned circuit, a local gate, and XX_β.

Our goal is to describe a certain *projection* of this polytope. Projection has the effect of introducing an existential quantifier into the above description: a point belongs to the projection of a polytope exactly when it is possible to extend the projected point by the discarded coordinates so that it satisfies the original constraints. This trades the actual data housed in the lost coordinates—which may be complicated to the point of distraction—for the mere predicate that such data exists. In our case, we seek to project away the coordinates (a_1, a_2, a_3), which leaves only constraints on (b_1, b_2, b_3), given in terms of $(\alpha_+, \alpha', \alpha'', \beta)$, ensuring that a prefix circuit of the indicated type *exists*, without actually naming it.

To compute this projection, we apply Fourier–Motzkin elimination to project away the remaining coordinates and eliminate redundancies in the resulting inequality set. These reduced inequality sets have the following form:

$$\begin{cases} \alpha_+ + \beta & \geq & b_1+b_2+b_3, \\ \min\left\{ \begin{array}{c} \alpha_+ + \beta - 2\alpha' \\ \alpha_+ + \beta - 2\beta \end{array} \right\} & \geq & -b_1+b_2+b_3, \\ \min\left\{ \begin{array}{c} \alpha_+ + \beta - \alpha' - \alpha'' \\ \alpha_+ + \beta - \alpha' - \beta \\ \alpha_+ + \beta - \beta - \alpha'' \end{array} \right\} & \geq & b_3, \end{cases}$$
$$\begin{cases} -\frac{\pi}{2} + \alpha_+ + \beta & \geq & -b_1+b_2+b_3; \\ \min\left\{ \begin{array}{c} \frac{\pi}{2} + \alpha_+ + \beta - 2\alpha' \\ \frac{\pi}{2} + \alpha_+ + \beta - 2\beta \end{array} \right\} & \geq & b_1+b_2+b_3, \\ \min\left\{ \begin{array}{c} \alpha_+ + \beta - \alpha' - \alpha'' \\ \alpha_+ + \beta - \alpha' - \beta \\ \alpha_+ + \beta - \beta - \alpha'' \end{array} \right\} & \geq & b_3, \end{cases}$$

where we have collected the inequalities which give communal upper bounds into single expressions using "min". Notationally absorbing β into the sequence of interaction strengths completes the proof.

See `check_main_xx_theorem` in `monodromy` [4] for an executable proof. □

Example 4.2. We include a visualization of an example XX–circuit polytope in Figure 4.

Remark 4.3. The two convex bodies in the statement of Theorem 4.1 are related by the linear transformation

$$(a_1, a_2, a_3) \mapsto \left(\frac{\pi}{2} - a_1, a_2, a_3\right).$$

Remark 4.4. Theorem 4.1 is manifestly invariant under permutation of the interaction strengths.

Remark 4.5. By dropping the assumption that the entries of positive canonical triples are ordered descending (as in Remark 2.8), we can rewrite the above inequality families in a manner that is more pleasingly symmetric. For example, the first[11] family is rewritten as:

$$\alpha_+ \geq a_+,$$
$$\min_k \alpha_+ - 2\alpha_k \geq \min_k a_+ - 2a_k,$$
$$\min_{k\neq\ell} \alpha_+ - \alpha_k - \alpha_\ell \geq \min_{k\neq\ell} a_+ - a_k - a_\ell.$$

We note that we have won these pleasing formulas by losing convexity: the "min"s appearing in the lower bounds encode disjunctions of linear sentences rather than conjunctions, so we see merely a non-convex union of these convex polytopes.

Remark 4.6 ([5]). In the case of a uniform interaction strength α, we compute the quantities appearing in the upper bounds of Theorem 4.1 to be

$$\alpha_+ = n\alpha,$$
$$\min_k \alpha_+ - 2\alpha_k = (n-2)\alpha,$$
$$\min_{k\neq\ell} \alpha_+ - \alpha_k - \alpha_\ell = (n-2)\alpha.$$

[11] The second is similar, but less pleasing to the eye.

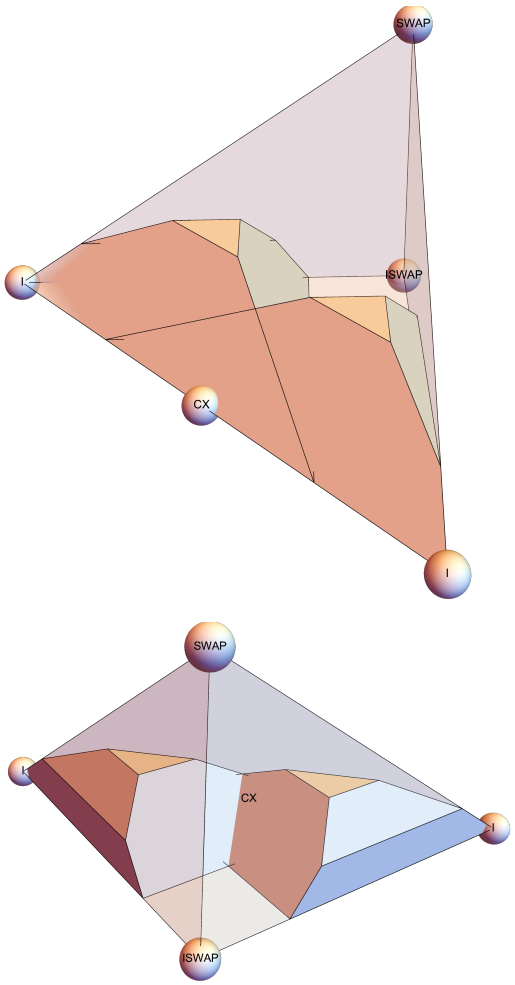


Figure 4: Two perspectives on the XX–circuit polytope for the interaction strength sequence $(\frac{\pi}{6}, \frac{\pi}{8}, \frac{\pi}{12})$. From the perspective where $SWAP$ is near the eyepoint, the strength faces are the pair facing inwards, the slant faces are the pair facing outwards, and the frustrum face is colored tan.

This causes the slant and frustrum inequalities to degenerate, which recovers a theorem of Zhang et al. as a special case.

5 The local theorem

In this section, we study the problem of appending a single new XX interaction strength β to a *specific circuit* formed from a sequence of strengths $(\alpha_1, \ldots, \alpha_n)$. Note that Theorem 4.1 gives us an understanding of the "global" effect of appending XX_β, where the interaction strengths are fixed but the circuit is allowed to range. Note also that if we are able to achieve such a local understanding, we would then like to use it in reverse: given a point p_{n+1} which Theorem 4.1 guarantees to be modelable using a circuit with strengths $(\alpha_1, \ldots, \alpha_n, \beta)$, we would like to guarantee the existence of—and algorithmically identify!—a point p_n which is modelable by $(\alpha_1, \ldots, \alpha_n)$ and for which p_{n+1} is reachable by appending XX_β and some local gates.

Excepting the caveat about algorithmic identification, this can be accomplished directly using nothing more than the methods of the monodromy polytope. However, because we are interested in circuit construction, we restrict what sorts of circuits we are willing to append to those of the particularly simple form given in Lemma 5.1. In trade, the method of the monodromy polytope no longer directly applies.

We show in Theorem 5.3 the "forward" direction of the strategy described above, then in Theorem 5.5 the "reverse" direction, culminating in the recursive step in a synthesis procedure whose full description we defer to Section 6. First, however, we introduce the simplified circuit which we will consider.

Lemma 5.1. *For any choice of a_1, a_2, β, d, e in*

$$U := CAN(a_1, a_2, 0) \cdot (Z_d \otimes Z_e) \cdot (XX_\beta),$$

there exist values r, s, t, u, b_1, and b_2 so that the operator U may be equivalently expressed as

$$(Z_r \otimes Z_s) \cdot CAN(b_1, b_2, 0) \cdot (Z_t \otimes Z_u).$$

Proof. The vector subspace

$$\mathfrak{g} = \text{span}\{XX, YY, XY, YX, ZI, IZ\}$$

forms a Lie subalgebra of $\mathfrak{pu}_4$, and the subspaces

$$\mathfrak{a} = \text{span}\{XX, YY\}, \qquad \mathfrak{k} = \text{span}\{IZ, ZI\}$$

give rise to a KAK decomposition yielding the desired result. □

Next, we note that this choice of simple local gates gives rise to the desired explicit expressions for the gate parameters.

Lemma 5.2. *Except for the outer parameters r, s, t, and u, the parameters in Lemma 5.1 are related by the equations*

$$\frac{c^2_{a_1-a_2}c^2_\beta + s^2_{a_1-a_2}s^2_\beta - c^2_{b_1-b_2}}{2c_{a_1-a_2}c_\beta s_{a_1-a_2}s_\beta} = c_{2(d+e)},$$
$$\frac{c^2_{a_1+a_2}c^2_\beta + s^2_{a_1+a_2}s^2_\beta - c^2_{b_1+b_2}}{2c_{a_1+a_2}c_\beta s_{a_1+a_2}s_\beta} = c_{2(d-e)}.$$

The outer parameters r, s, t, and u can then be deduced from a linear system with input the phases of the top half of the left-hand matrix.

Proof. The trigonometric equalities follow by equating the square-norms of the matrix entries in Lemma 5.1. The $(1,1)$ and $(3,3)$ entries respectively yield

$$|c_{b_1-b_2}|^2 = |c_\beta c_{a_1-a_2} - e^{2i(d+e)}s_\beta s_{a_1-a_2}|^2,$$
$$|c_{b_1+b_2}|^2 = |c_\beta c_{a_1+a_2} - e^{-2i(d-e)}s_\beta s_{a_1+a_2}|^2,$$

where we have used the absolute values to suppress some of the phases. We then apply the identity

$$|x + re^{i\theta}|^2 = x^2 + r^2 + 2xr\cos\theta$$

and isolate d and e to deduce the statement.

The linear system then arises by inspecting the phases of any nondegenerate quadruple of entries. For example, the nonzero entries in the top half, read left-to-right, have respective phases

$$\exp(-i(r+s+t+u)),$$
$$\exp(-i(r-s+t+u),$$
$$-i\exp(-i(-r+s-t+u)),$$
$$-i\exp(-i(r+s-t-u)).$$

This collection of linear combinations is of full rank. □

We can interpret the constraints imposed by these expressions on the positive canonical triples in terms of β.

Theorem 5.3 ("Interference inequalities")**.** *For positive canonical triples $(a_1,a_2,0)$ and $(b_1,b_2,0)$ and for β an interaction strength, there exist parameters d and e satisfying*

$$CAN(a_1,a_2,0)\cdot(Z_d\otimes Z_e)\cdot XX_\beta \equiv CAN(b_1,b_2,0)$$

if and only if the following inequalities hold:[12]

$$a_1+a_2-\beta \le b_1+b_2 \le \frac{\pi}{2} - \left|\frac{\pi}{2} - (a_1+a_2+\beta)\right|,$$
$$|a_1-a_2-\beta| \le b_1-b_2 \le a_1-a_2+\beta.$$

Moreover, the local gates witnessing the equivalence can be taken to be Z–rotations.

[12]It is extremely unusual that image under Π of a circuit with constrained local gates is again a polytope. This is, perhaps, the most important ingredient in our approach.

Proof. Starting from Lemma 5.2, there exist solutions to d and e exactly when the following inequalities are met:

$$|c^2_{a_1-a_2}c^2_\beta + s^2_{a_1-a_2}s^2_\beta - c^2_{b_1-b_2}| \le |2c_{a_1-a_2}c_\beta s_{a_1-a_2}s_\beta|,$$
$$|c^2_{a_1+a_2}c^2_\beta + s^2_{a_1+a_2}s^2_\beta - c^2_{b_1+b_2}| \le |2c_{a_1+a_2}c_\beta s_{a_1+a_2}s_\beta|.$$

Using the inequalities $a_1+a_2 \le \frac{\pi}{2}$, $a_1 \ge a_2$, and $0 \le \beta \le \frac{\pi}{4}$, we see that both the right-hand quantities are always positive, hence the right-hand absolute value can be suppressed. The left-hand absolute value can be equivalently expressed as a pair of inequalities, giving

$$-2c_{a_1-a_2}c_\beta s_{a_1-a_2}s_\beta \le c^2_{a_1-a_2}c^2_\beta + s^2_{a_1-a_2}s^2_\beta - c^2_{b_1-b_2},$$
$$c^2_{a_1-a_2}c^2_\beta + s^2_{a_1-a_2}s^2_\beta - c^2_{b_1-b_2} \le 2c_{a_1-a_2}c_\beta s_{a_1-a_2}s_\beta,$$
$$-2c_{a_1+a_2}c_\beta s_{a_1+a_2}s_\beta \le c^2_{a_1+a_2}c^2_\beta + s^2_{a_1+a_2}s^2_\beta - c^2_{b_1+b_2},$$
$$c^2_{a_1+a_2}c^2_\beta + s^2_{a_1+a_2}s^2_\beta - c^2_{b_1+b_2} \le 2c_{a_1+a_2}c_\beta s_{a_1+a_2}s_\beta.$$

Factoring the quadratics then yields

$$(c_{a_1-a_2}c_\beta - s_{a_1-a_2}s_\beta)^2 \le c^2_{b_1-b_2},$$
$$c^2_{b_1-b_2} \le (c_{a_1-a_2}c_\beta + s_{a_1-a_2}s_\beta)^2,$$
$$(c_{a_1+a_2}c_\beta - s_{a_1+a_2}s_\beta)^2 \le c^2_{b_1+b_2},$$
$$c^2_{b_1+b_2} \le (c_{a_1+a_2}c_\beta + s_{a_1+a_2}s_\beta)^2.$$

Rewriting the binomials as cosines of differences / sums and then converting square cosines to double-angle cosines yields

$$c_{2(a_1-a_2+\beta)} \le c_{2(b_1-b_2)} \le c_{2(a_1-a_2-\beta)},$$
$$c_{2(a_1+a_2+\beta)} \le c_{2(b_1+b_2)} \le c_{2(a_1+a_2-\beta)}.$$

Finally, we use the piecewise monotonicity and reflection invariance of cosine, as well as the bounds on the inputs, to deduce inequalities on the angles:

$$2(a_1-a_2+\beta) \ge 2(b_1-b_2),$$
$$2(b_1-b_2) \ge \max\{2(a_1-a_2-\beta), -2(a_1-a_2-\beta)\},$$
$$\min\{2(a_1+a_2+\beta), 2\pi - 2(a_1+a_2+\beta)\} \ge 2(b_1+b_2).$$
$$2(b_1+b_2) \ge 2(a_1+b_2-\beta).$$

Linear rearrangement yields the claimed inequality family. □

Example 5.4. In Figure 5, we give a visualization of the regions accessible via Theorem 5.3.

Theorem 5.5 (cf. Figure 6)**.** *Given a positive canonical triple (b_1,b_2,b_3) satisfying the conditions of Theorem 4.1 for a sequence of interaction strengths $(\alpha_1,\ldots,\alpha_n,\beta)$, there always exists a positive canonical triple (a_1,a_2,a_3) satisfying the conditions of Theorem 4.1 for the sequence $(\alpha_1,\ldots,\alpha_n)$ and for which*

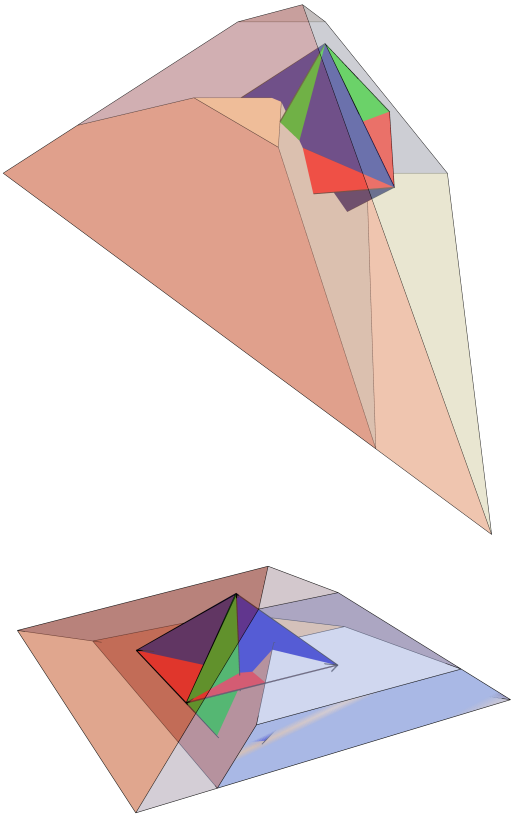


Figure 5: Two perspectives on the configuration of polytopes in Theorem 5.3. The solid, inner polytope is the XX–circuit polytope for a sequence of interaction strengths $(\alpha_1, \ldots, \alpha_n)$, and the translucent outer polytope is the XX–circuit polytope for the sequence $(\alpha_1, \ldots, \alpha_n, \beta)$. We have chosen a particular point on the edge of the inner figure and drawn the regions accessible from it by applying Theorem 5.3 for the interaction strength β to any pair of coordinates (red: XX and YY; green: YY and ZZ; blue: XX and ZZ). These appear as a triple of flat polygons. Theorem 5.5 states that when the point on the inner body is permitted to range, the union of the corresponding translates of the red, green, and blue polygons sweeps out the entirety of the outer body.

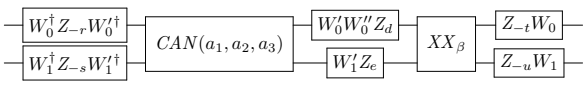


Figure 6: The circuit emitted by a typical single step of Procedure 6.1 (i.e., by Theorem 5.5), modeling $CAN(b_1, b_2, b_3)$ in terms of $CAN(a_1, a_2, a_3)$, XX_β, and local gates. The gates W, W', W'' are quarter-turns which realize the action of Weyl group elements on $a, b \in \mathfrak{A}_{C_2}$.

there are Weyl reflections w, w' so that the following is solvable:

$$CAN(a_1, a_2, a_3)^w \cdot (Z_d \otimes Z_e) \cdot CAN(\beta) \equiv CAN(b_1, b_2, b_3)^{w'}.$$

The outer gates witnessing the local equivalence can be taken to be Z-rotations.

Proof. Any canonical gate $CAN(b_1, b_2, b_3)$ can be written as

$$CAN(b_1, b_2, b_3) = CAN(0, 0, b_3) \cdot CAN(b_1, b_2, 0).$$

Applying Theorem 5.3 to the right factor gives

$$\begin{aligned} CAN(b_1, b_2, b_3) = CAN(0, 0, b_3) &\cdot \\ &\cdot (Z_{-r} \otimes Z_{-s}) \cdot CAN(a_1, a_2, 0) \cdot \\ &\cdot (Z_d \otimes Z_e) \cdot CAN(\beta) \cdot \\ &\cdot (Z_{-t} \otimes Z_{-u}), \end{aligned}$$

under certain conditions on a_1, a_2, b_1, b_2, and β. Since local Z–rotations commute with canonical gates of the form $CAN(0, 0, b_3)$, we may abbreviate this to

$$\begin{aligned} CAN(b_1, b_2, b_3) = (Z_{-r} \otimes Z_{-s}) \cdot CAN(a_1, a_2, b_3) &\cdot \\ &\cdot (Z_d \otimes Z_e) \cdot CAN(\beta) \cdot \\ &\cdot (Z_{-t} \otimes Z_{-u}). \end{aligned}$$

Additionally, our choice to factor out b_3 is immaterial: there are Weyl reflections which permute the coordinates within a canonical triple, so by conjugating $CAN(b_1, b_2, b_3)$ we can place any of the three values in the final slot. In short, we may appeal to Theorem 5.3, provided we fix one coordinate and potentially disorder the positive canonical triples.

From here, our proof strategy is similar to that of Theorem 4.1. Theorem 4.1 itself furnishes us with linear constraints on the spaces of triples (b_1, b_2, b_3) so that a triple satisfies the constraints if and only it can be realized as the positive canonical coordinate of an XX–circuit with interaction strengths $(\alpha_1, \ldots, \alpha_n, \beta)$. Rather than working with ordered triples (a_1, a_2, a_3), we instead consider unordered triples (a_h, a_ℓ, a_f)—to be referred to as the "high", "low", and "fixed" coordinates—as in Remark 4.5. Then, we interrelate the a– and b–coordinates:

- We select one coordinate b_f from (b_1, b_2, b_3) to serve as the "fixed" coordinate (and take the union over such choices), and we set $a_f = b_f$.
- On a_h and a_ℓ, we impose the constraint $a_h \geq a_\ell$. Similarly, of the remaining coordinates in (b_1, b_2, b_3), we pick b_h to be the larger and b_ℓ to be the smaller.
- We impose the constraints from Theorem 5.3 on $(a_h, a_\ell, 0)$, $(b_h, b_\ell, 0)$, and β.

Let us call the resulting (nonconvex) polytope P. Points in P capture the following interrelated pieces of data:

- A canonical coordinate (b_1, b_2, b_3) which admits expression as an XX–circuit with interaction strengths $(\alpha_1, \ldots, \alpha_n, \beta)$.
- A canonical coordinate (a_1, a_2, a_3) which admits expression as an XX–circuit with interaction strengths $(\alpha_1, \ldots, \alpha_n)$.
- A choice of value to share among the a– and b–coordinates.
- The condition that, among the unshared coordinates, there exists a circuit of the form in Lemma 5.1 relating them. (As in the first two bullets, the polytope does *not* record the literal data of such a circuit, only the predicate that one exists.)

By projecting away (a_h, a_ℓ, a_f) from P, we produce the polytope of positive canonical triples (b_1, b_2, b_3) which can be expressed as XX–circuits with the specified interaction strengths, together with the predicate constraint that the last step in the circuit decomposition can be written in the form of the Theorem statement. This is a subpolytope of that of Theorem 4.1, which merely tracks positive canonical triples which can be expressed as XX–circuits with the specified interaction strengths, *without* the constraint on the final local operator. Appealing again to a computer algebra system, we find that these two polytopes are equal.

See `regenerate_xx_solution_polytopes` in `monodromy` [4] for an executable proof. □

Remark 5.6. Naively specified, the polytope P in the proof of Theorem 5.5 has many convex components: the two convex regions of a– and b–coordinates each contribute factors of 2, the choice of which coordinate to fix contributes a factor of 3, and the choice of which slant and frustrum bounds apply to the disordered a–coordinates contribute factors of 2 and 3. However, the projection of P onto the b–coordinates, which we used to conclude the theorem, can be shown to have only four regions:

- The choice of convex region of b–coordinates is free, but one then uses the same choice for a–coordinates.
- The fixed coordinate a_f is taken to be either b_1 or b_3.
- For the unreflected (resp., reflected) convex region of b–coordinates, the slant (resp., strength) inequality is imposed either on a_f or a_h depending on whether $a_f = b_1$ or $a_f = b_3$.
- The frustrum bound is always imposed on a_ℓ.

The inequalities describing these regions are given in Figure 20.

Remark 5.7. It is possible for the technophobic reader to rearrange the proofs of Theorem 4.1 and Theorem 5.5 so as to avoid computer algebra systems. First, break Theorem 4.1 into a forward implication, that the positive canonical triple associated to an XX–circuit satisfies the indicated inequality set, and the reverse implication. The forward implication can be checked by hand, using a judiciously chosen subset of inequalities from the monodromy polytope; the reverse implication is much harder from this point of view, so we set it aside for a moment.

Now we turn to Theorem 5.5. Its proof also relies on a computer algebra system, but we may severely limit the amount of work by inspecting only the convex summands described in Remark 5.6, which is then small enough to accomplish manually. With only the forward implication of Theorem 4.1 established, the proof of Theorem 5.5 instead shows that if the b–coordinate belongs to the polytope named by Theorem 4.1 for $(\alpha_1, \ldots, \alpha_n, \beta)$, then there exists an a–coordinate in the polytope named by Theorem 4.1 for $(\alpha_1, \ldots, \alpha_n)$ which is related to the b–coordinate by a particular single-step XX–circuit. Following the induction described in Procedure 6.1 then yields the missing reverse implication of Theorem 4.1, which in turn yields the full strength of Theorem 5.5.

6 Optimal synthesis

We now put the pieces together to form an optimal synthesis routine. The actual synthesis process is now straightforward, given in Procedure 6.1, but it is trickier to pin down exactly what is meant by "optimal". For instance, the notion of optimality considered by Zhang et al. [5] is to minimize two-qubit operation count—but in a larger gateset, where different gates may have uneven performance impact, optimizing count alone may not optimize performance. Relatedly, if performance is the true goal and the performance penalty incurred for using gates is high, it may be preferable to synthesize a circuit modeling some canonical triple $a' \neq a$ which requires fewer gates, trading the performance hit due to the mismatch for performance gain of dropping some of the gates.

Let us begin with the synthesis procedure itself:

Procedure 6.1 (cf. Figure 6)**.** The existence claim of Theorem 5.5 can be promoted into an algorithmically effective synthesis routine. Given a sequence of interaction strengths $(\alpha_1, \ldots, \alpha_n, \beta)$ and a positive canonical triple (b_1, b_2, b_3) which belongs to the associated circuit polytope, the polytope P from the proof of Theorem 5.5 can then be specialized so that only a_h and a_ℓ are free variables. (We report these inequality sets in Figure 21.) The content of Theorem 5.5 is that this specialization is always nonempty, so we may find

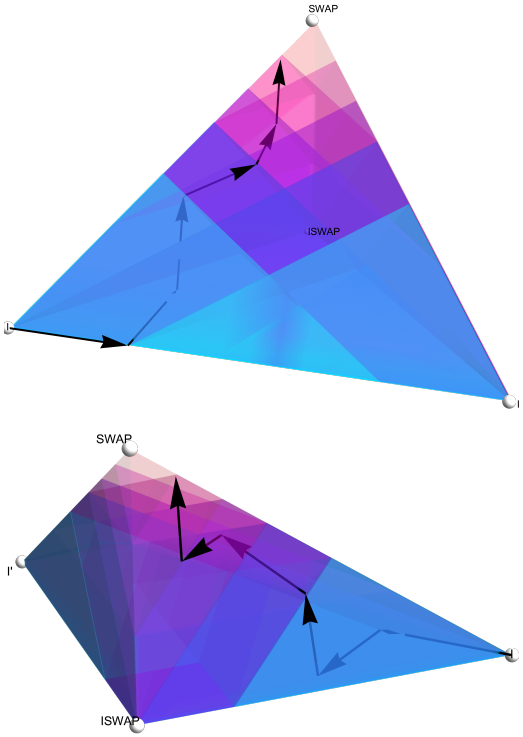


Figure 7: Path of intermediate points in $\mathfrak{A}_{C_2}$ produced by Procedure 6.1 synthesizing $CAN(0.707, 0.687, 0.687)$ into an XX–circuit with strength sequence $(\frac{\pi}{8}, \frac{\pi}{8}, \frac{\pi}{12}, \frac{\pi}{12}, \frac{\pi}{12}, \frac{\pi}{12})$. The various colored regions are the circuit polyhedra for truncations of this sequence of interaction strengths.

a point (a_h, a_ℓ, a_f) in it (e.g., by calculating line-line intersections until we produce a vertex). This pair of points can then be fed to Lemma 5.2, which produces the angle values for the Z–rotations. This proceeds recursively until the sequence of interaction strengths is exhausted.

Example 6.2. In Figure 7, we include a visualization of the intermediate steps produced when using Procedure 6.1 to synthesize an XX–circuit for a certain canonical point against a particular sequence of interaction strengths.

To progress, we need a quantitative definition of optimality.

Definition 6.3. Given a target unitary U, a gate set $\mathcal{S}$, and a cost function $\mathcal{C}_\mathcal{S}$ which consumes U and an $\mathcal{S}$–circuit C, the *approximate synthesis task* is to produce an $\mathcal{S}$–circuit C maximizing $\mathcal{C}_\mathcal{S}(U, C)$. Cost functions can enjoy a variety of pleasant properties:

***Separable*:** For a circuit template $C(\theta)$, $\mathcal{C}_\mathcal{S}(U, C)$ can be written as a sum $\mathcal{C}'(U, C(\theta)) + \mathcal{C}''_\mathcal{S}(C, \theta)$ where $\mathcal{C}'$ depends only on the unitary which the circuit models and $\mathcal{C}''$ depends only on the circuit and parameters—but not its relationship to U.

***Locally invariant*:** $\mathcal{C}''_\mathcal{S}(C, \theta) = \mathcal{C}''_\mathcal{S}(C)$ is invariant under choice of parameters θ for local gates in the circuit C.

***Monotonic*:** Suppose that $\mathcal{C}$ is a separable cost function. If C is contained as a subcircuit in D, then $\mathcal{C}''_\mathcal{S}(C, \theta) \leq \mathcal{C}''_\mathcal{S}(D, (\theta, \phi))$.

***Non-approximating*:** The separable cost function $\mathcal{C}$ has $\mathcal{C}'$ given by

$$\mathcal{C}'(U, V) = \begin{cases} 0 & \text{when } U = V, \\ \infty & \text{otherwise.} \end{cases}$$

These features are chosen both because they feed into an efficient algorithm for optimal synthesis and because they are satisfied in the following guiding example:

Example 6.4. The *average infidelity* of two gates U and V is

$$\begin{aligned} I(U, V) &= 1 - \int_{\psi \in \mathbb{P}(\mathbb{C}^4)} \langle \psi | U^\dagger V | \psi \rangle^2 \\ &= \frac{16 - |\operatorname{tr} U^\dagger V|^2}{4 \cdot 5} \in [0, 4/5]. \end{aligned}$$

For $\mathcal{S}$ a finite collection of XX–interactions with costs $c\colon \mathcal{S} \to \mathbb{R}$, we define a separable, locally invariant, monotonic cost function by

$$\mathcal{C}_\mathcal{S}(U, C) = I(U, C) + \sum_{s \in C} c(s).$$

Average gate infidelity satisfies a few pleasant generic properties, but it is also tightly connected to the theory of KAK decompositions. We record these properties below.

Remark 6.5. Average infidelity detects gate equivalence, in the sense that $I(U, V) = 0$ if and only if $U = V$. It is also symmetric: $I(U, V) = I(V, U)$. However, it fails to satisfy the triangle inequality, even when U and V belong to the canonical family, hence does *not* give a metric. It satisfies compositionality only to first order:

$$\begin{aligned} &16 - 20 \cdot I(UU', (U + \varepsilon E)(V + \zeta F)) \\ &\quad = |\operatorname{tr} V^\dagger U^\dagger (U + \varepsilon E)(V + \zeta F)|^2 \\ &\quad = |1 + \varepsilon \operatorname{tr} U^\dagger E + \zeta \operatorname{tr} V^\dagger F + \varepsilon\zeta \operatorname{tr} V^\dagger U^\dagger E F|^2. \end{aligned}$$

Lemma 6.6 ([8])**.** *Let $U = CAN(a_1, a_2, a_3)$ and $V = CAN(b_1, b_2, b_3)$ be two canonical gates with parameter differences $\delta_j = (a_j - b_j)$. Their average gate infidelity is given by*

$$20 \cdot I(U, V) = 16 - 16 \left(\prod_j \cos^2 \frac{\delta_j}{2} + \prod_j \sin^2 \frac{\delta_j}{2} \right). \quad \square$$

Lemma 6.7 ([18])**.** *Suppose that C_1, C_2 are fixed canonical gates and that L_1, L_1' are fixed local gates. Letting L_2 and L_2' range over all local gates, the value $I(L_1 C_1 L_1', L_2 C_2 L_2')$ is minimized when taking $L_2 = L_1$ and $L_2' = L_1'$.* $\square$

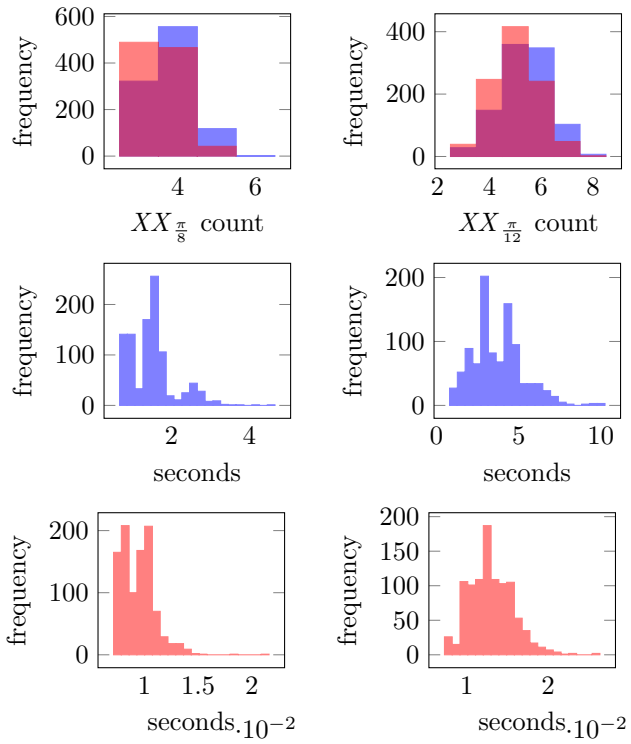


Figure 8: Comparison of the output and wall-time characteristics of our algorithm (in red) and that of a `numpy` numerical search (in blue), when targeting the two-qubit gate sets $\mathcal{S} = \{XX_{\frac{\pi}{8}}\}$ and $\mathcal{S} = \{XX_{\frac{\pi}{12}}\}$. We plot the histograms of wall-times separately, as numerical search is $> 200\times$ slower. Numerical search can fall in local wells and produce suboptimal circuits.

We now describe an optimal synthesis procedure for a nice cost function:

Procedure 6.8. Let $\mathcal{C}$ be a separable, locally invariant, and monotonic cost function. Let $\mathcal{S}$ be a finite gate set of XX–type interactions, and consider the set of circuit templates given by interleaving unconstrained local gates into the words formed from $\mathcal{S}$. Traverse these available circuit templates (i.e., the words in $\mathcal{S}$) by ascending order of $\mathcal{C}''_{\mathcal{S}}$.[13] For each such circuit template C, use Theorem 4.1 to calculate the circuit polytope $\Pi(C)$. Calculate the point $p \in \Pi(C)$ which optimizes $\mathcal{C}'(U, CAN(p))$. If the total cost $\mathcal{C}_{\mathcal{S}}$ is the best seen so far, retain C and p. Continue to traverse circuit templates until $\Pi(U) \in \Pi(C)$, at which point $\mathcal{C}'$ vanishes and the ordering of circuit templates guarantees that all future circuit templates will yield a worse cost.[14] Finally, apply Procedure 6.1 to synthesize a C–circuit for $CAN(p)$, then apply Lemma 6.7 to produce U itself.

Remark 6.9. In Figure 8, we study the execution characteristics of Procedure 6.8 compared to those of blind numerical search. The implementation of our method is available in Qiskit's `quantum_info` subpackage as the class `XXDecomposer` [3]. Given a Haar-randomly chosen two-qubit unitary operator U, our numerical search procedure is to let `numpy`'s generic optimizer explore the space of circuits of a particular depth, with the objective of minimizing the infidelity with U. If the optimizer cannot find a circuit with infidelity below some threshold, we retry with a circuit of the next larger depth. Altogether, this is similar to what is implemented in `NuOp` [6], among other compilation suites. The histograms reported in Figure 8 are the result of sampling over many such U, targeting either the gate set $\mathcal{S} = \{XX_{\frac{\pi}{8}}\}$ or $\mathcal{S} = \{XX_{\frac{\pi}{12}}\}$.[15]

It remains to describe how to find the point $p \in \Pi(C)$ which optimizes $\mathcal{C}'(U, CAN(p))$. For a non-approximating cost function, this can be probed directly: if $\Pi(U) \in \Pi(C)$, then we take $p = \Pi(U)$, and otherwise we reject $\Pi(C)$ entirely. For the approximating cost function defined in Example 6.4, we use the following more elaborate result:

Theorem 6.10. *Let P be an XX–circuit polytope, and let $F \subseteq P$ be an open facet within it. For $p \in \mathfrak{A}_{C_2}$ a fixed positive canonical triple, if $b = q$ is a critical point of the infidelity distance $I|_{a=p,b\in F}$ as constrained to F, then q is also a critical point of the Euclidean distance to p as constrained to F.*

Proof. Theorem 4.1 gives an explicit enumeration of the available open facets of P, and we approach the optimization problem over each facet separately. We defer this to Appendix A. □

This result means that we can repurpose the standard procedure used to calculate the nearest point in Euclidean distance to instead find the best approximating canonical triple. Namely, to calculate the nearest point in Euclidean distance, project the point onto the affine subspaces spanned by each facet of the polytope (e.g., by solving a least-squares problem), retain those projections which belong to the polytope, and from that finite set select the point of minimum (infidelity) distance.

Remark 6.11. This is extremely unusual behavior for these two optimization problems and relies on the specific form of the polytopes appearing in Theorem 4.1. For contrast, consider the line passing through the origin with slope $(\frac{\pi}{4}, \frac{\pi}{50}, \frac{\pi}{50})$ and the off-body point $(\frac{83\pi}{400}, \frac{83\pi}{400}, \frac{83\pi}{400})$. The fidelity-nearest point appears after traveling for one unit of time, but the Euclidean-nearest point appears after traveling for $\approx 95\%$ of a unit of time.

[13] Using a priority queue, one can perform this traversal without enumerating all possible words beforehand.

[14] Theorem 4.1 guarantees that this termination condition will eventually be met provided $\mathcal{S}$ contains any interaction XX_β with $\beta \in (0, \frac{\pi}{2})$.

[15] Neither our implementation of Procedure 6.8 nor our invocation of `numpy` is particularly clever. We expect that both distributions can be shifted left with further optimization of the implementations, but that the multiplicative difference will be at least as large between "optimal" implementations of each synthesis method.

Remark 6.12. Numerical experiment indicates that the nearest point under infidelity distance exactly agrees with the nearest point under Euclidean distance—i.e., that the same critical point achieves the minimum value in both of these searches. However, this conjecture yields no algorithmic speedup when producing these minimizers, so we are not motivated to pursue it here.

7 Gateset optimization and numerical experiment

In this section, we bring the theory of Section 6 to bear on deciding which native gates are worth bestowing on a device. Even if a device is physically capable of enacting some quantum operation, there is calibration overhead to making that operation available as a reliable user-facing gate. At the same time, the more high-fidelity native interactions are available, the more clever and adaptable a synthesis method (including ours) can be. Accordingly, we would like to find a small set of XX operations that optimizes certain objective functions which measure synthesis performance. The primary objective with which we will concern ourselves is expected cost:

Definition 7.1. For a two-qubit unitary $U \in PU(4)$ and a native gate set $\mathcal{S}$, let $\mathcal{C}_{\mathcal{S}}(U) := \min_C \mathcal{C}_{\mathcal{S}}(U, C)$ be a cost function as in Definition 6.3 (e.g., Example 6.4 or its non-approximating variant). The *expected cost* is defined as

$$\langle \mathcal{C}_{\mathcal{S}} \rangle = \int_{U \in PU(4)} \mathcal{C}_{\mathcal{S}}(U)\, d\mu^{\text{Haar}}.$$

For XX–based gate sets $\mathcal{S}$ and for favorable cost functions, we now show how to compute this value *exactly*. Starting with the definition

$$\langle \mathcal{C}_{\mathcal{S}} \rangle = \int_{U \in PU(4)} \min_C \mathcal{C}_{\mathcal{S}}(U, C)\, d\mu^{\text{Haar}},$$

we use separability and non-approximation to reduce to the case where U admits an exact model by C:

$$\langle \mathcal{C}_{\mathcal{S}} \rangle = \int_{U \in PU(4)} \min_{U = C(\theta)} \mathcal{C}''_{\mathcal{S}}(C, \theta)\, d\mu^{\text{Haar}}.$$

By assuming $\mathcal{S}$ finite and $\mathcal{C}''_{\mathcal{S}}$ locally invariant, we learn that the integrand $\min_{U \in C} \mathcal{C}''_{\mathcal{S}}(C)$ takes on finitely many values, supported by finitely many choices of C. By sorting the C compatibly with $\mathcal{C}''_{\mathcal{S}}(C)$, we may further reduce to

$$\langle \mathcal{C}_{\mathcal{S}} \rangle = \sum_C \int_{\substack{U \in \text{Image}(C) \\ U \notin \text{Image}(C' | C' < C)}} \mathcal{C}''_{\mathcal{S}}(C)\, d\mu^{\text{Haar}}.$$

Since $\mathcal{C}''_{\mathcal{S}}(C)$ is constant on each region, each summand is given by the reweighted Haar volume of the corresponding region. Since constant functions pull back from constant functions, we can also push these integrals forward along Π and compute them in $\mathfrak{A}_{C_2}$:

$$\langle \mathcal{C}_{\mathcal{S}} \rangle = \sum_C \mathcal{C}''_{\mathcal{S}}(C) \left(\Pi_* \mu^{\text{Haar}} \left(\Pi(C) \setminus \bigcup_{C' < C} \Pi(C') \right) \right).$$

Altogether, this reduces the problem to calculating the Haar volume of the polytopes which appear in Theorem 4.1. A formula for $\Pi_* \mu^{\text{Haar}}$ which enables this was previously reported by Watts et al.:

Lemma 7.2 ([19], [20], [2])**.** *The pushforward of the Haar measure is given by*[16,17]

$$\Pi_* d\mu^{\text{Haar}} = \frac{384}{\pi} \prod_{1 \le j < k \le 3} \sin(2c_j + 2c_k) \sin(2c_j - 2c_k).$$

□

Such trigonometric integrals over tetrahedra can be performed exactly. Altogether, this gives us quantitative means by which to study the effect of tuning the inputs to a parametric gate set, e.g., $\mathcal{S}(x) = \{XX_{\frac{\pi}{4}}, XX_x\}$. A parametric choice of gate set requires a parametric cost function, and our parametric cost function of interest is as follows:

Definition 7.3. In our setting, we find it experimentally justified to assume an *affine error model*: we take XX_x to have fidelity cost $mx + b$ for some experimentally determined values of m and b.[18] From this,

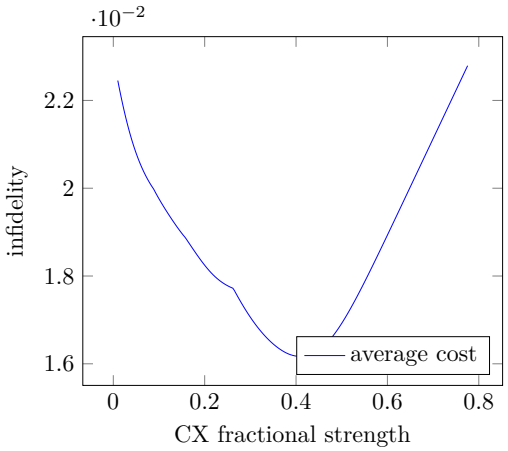


Figure 9: The expected infidelity of a Haar-random operator decomposed exactly into $\mathcal{S}_x = \{CX, XX_x\}$. Uses an additive affine error model with offset $b \approx 1.909 \times 10^{-3}$ and slope $\frac{\pi}{4} \cdot m \approx 5.76 \times 10^{-3}$.

[16] The extra factor of 2 appearing in this formula comes from a different scaling of our coordinate systems.

[17] This density function has a unique local maximum at $(\frac{\pi}{4}, \frac{\pi}{8}, 0)$.

[18] In one experiment, we measured $\frac{\pi}{4} \cdot m \approx 5.76 \times 10^{-3}$ and $b \approx 1.909 \times 10^{-3}$. This reported offset b incorporates the average infidelity cost of local post-rotations, so as to better model the total circuit execution cost while maintaining local invariance.

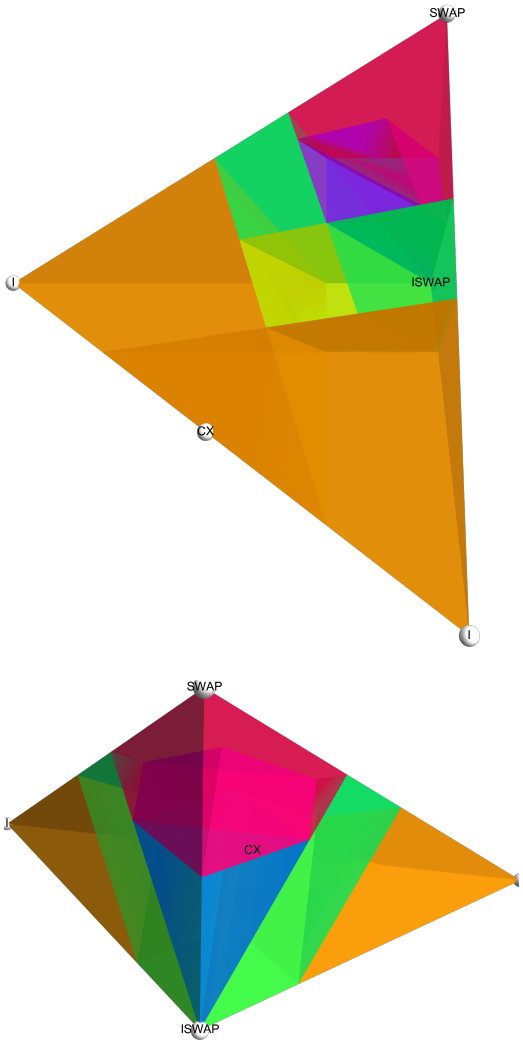


Figure 10: An optimal set of $\mathcal{S}$–circuit polytopes covering $\mathfrak{A}_{C_2}$ for $\mathcal{S} = \{XX_{\frac{\pi}{4}}, XX_{\frac{\pi}{8}}\}$. There are six regions depicted: $(\frac{\pi}{8}, \frac{\pi}{8}, \frac{\pi}{8})$ in orange, $(\frac{\pi}{8}, \frac{\pi}{8}, \frac{\pi}{4})$ in yellow, $(\frac{\pi}{8}, \frac{\pi}{8}, \frac{\pi}{8}, \frac{\pi}{8})$ in green, $(\frac{\pi}{8}, \frac{\pi}{4}, \frac{\pi}{4})$ in blue, $(\frac{\pi}{8}, \frac{\pi}{8}, \frac{\pi}{8}, \frac{\pi}{4})$ in purple, and $(\frac{\pi}{4}, \frac{\pi}{4}, \frac{\pi}{4})$ in red. There are also six regions which have circuit depth at most two, hence they do not contribute volume and we suppress them from the picture.

we build a separable, locally invariant, additive cost component by

$$\mathcal{C}''_{\mathcal{S}}(C) = \sum_{XX_x \in C} (mx + b).$$

Remark 7.4. The reader who would like to account, in the above framework, for the *worst case* cost of the interleaved single-qubit operations can absorb that extra amount into the b parameter.

Example 7.5. Consider the gate set $\mathcal{S} = \{XX_{\frac{\pi}{4}}, XX_x\}$ with cost given by the additive affine error model with parameters $b \approx 1.909 \times 10^{-3}$ and $\frac{\pi}{4} \cdot m \approx 5.76 \times 10^{-3}$. In Figure 9, we display how the expected infidelity of synthesizing an $\mathcal{S}$–circuit for a Haar-randomly chosen unitary varies with the gate set parameter x. The ends of this curve degenerate to the case of the smaller

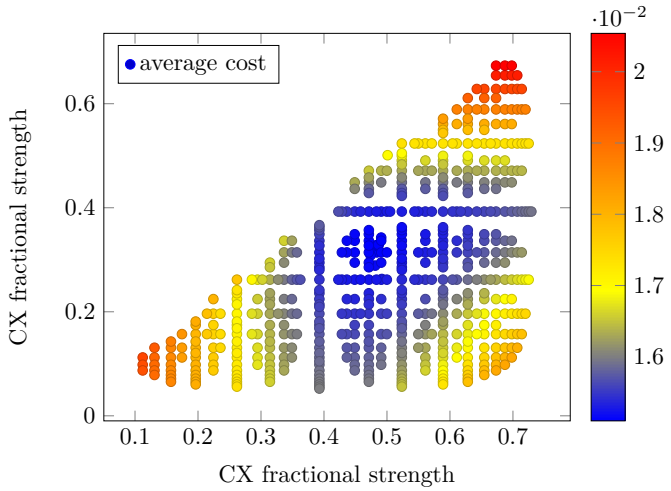


Figure 11: The expected infidelity of a Haar-random operator decomposed exactly into $\mathcal{S}_{x,y} = \{CX, XX_x, XX_y\}$. Uses an additive affine error model with offset $b \approx 1.909 \times 10^{-3}$ and slope $\frac{\pi}{4} \cdot m \approx 5.76 \times 10^{-3}$.

gate set $\{XX_{\frac{\pi}{4}}\}$. The precise location of the optimum in the middle depends on the ratio m/b; for experimentally realistic error models like the one depicted here, it is located near $\frac{\pi}{8}$, achieving an expected infidelity of 1.62×10^{-2}. We observe also that the basin for this minimum is fairly wide, so that $\frac{\pi}{8}$ is a good choice for inclusion in a native gate set even if the error model varies somewhat over time or across a device.[19] Finally, in Figure 10 we depict the optimal synthesis regions within the Weyl alcove for the gate set $\{XX_{\frac{\pi}{4}}, XX_{\frac{\pi}{8}}\}$.

Example 7.6. Consider next the gate set $\mathcal{S} = \{XX_{\frac{\pi}{4}}, XX_x, XX_y\}$ with the same cost function. In Figure 11, we display the expected infidelity of synthesizing an $\mathcal{S}$–circuit for a Haar-randomly chosen unitary against both parameters x and y. The edges of this triangular figure degenerate to the case discussed in Example 7.5 along the lines $x = \frac{\pi}{4}$, $y = 0$, and $x = y$. As before, the precise location of the optimum in the middle depends on the ratio m/b, but for experimentally realistic error models like the one depicted here, it is located near $(x, y) = (\frac{\pi}{8}, \frac{\pi}{12})$, this time achieving an expected infidelity of 1.51×10^{-2}. Again, the basin is fairly wide and the minimum fairly independent of the value of m/b, so that $(\frac{\pi}{8}, \frac{\pi}{12})$ are good choices for inclusion in a native gate set even if the observed error model exhibits mild variation over time or across a device. In Figure 12 we depict the optimal synthesis regions within the Weyl alcove for the gate set $\{XX_{\frac{\pi}{4}}, XX_{\frac{\pi}{8}}, XX_{\frac{\pi}{12}}\}$.

Example 7.7. Taking these results for exact synthesis as inspiration, we can also explore effects introduced by approximate synthesis. Our results here cannot be so clean, because we lose access to our

[19] Low-denominator rational multiples of $\frac{\pi}{4}$ are also easier to use in a randomized benchmarking scheme.

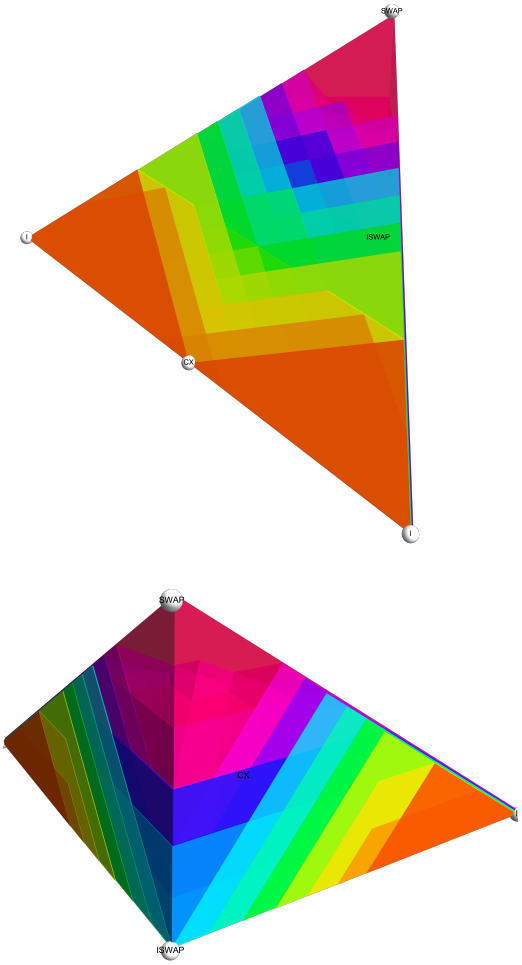


Figure 12: An optimal set of $\mathcal{S}$–circuit polytopes covering $\mathfrak{A}_{C_2}$ for $\mathcal{S} = \{XX_{\frac{\pi}{4}}, XX_{\frac{\pi}{8}}, XX_{\frac{\pi}{12}}\}$. The nineteen regions are too many to name explicitly, but their hues indicate an increasing cost from a minimum at I to a maximum at $SWAP$. There are also ten regions which have circuit depth at most two, hence they do not contribute volume and we suppress them from the picture.

method for analytic calculation, but we can still perform Monte Carlo experiments to analyze the relationship between $\mathcal{S}_x = \{CX, XX_x\}$ and the expected infidelity. The plot in Figure 13 shares many of the same qualitative features as Figure 9 (e.g., the approximate position of the global minimum, and the non-concave kink near $x = \frac{\pi}{2} \cdot 1/3$), with an overall vertical shift coming from the approximation savings. The global optimum for approximate synthesis into $\mathcal{S}_{x,y} = \{CX, XX_x, XX_y\}$ is again near to the global optimum for exact synthesis, so we re-use the user-friendly value of $(x, y) = (\frac{\pi}{8}, \frac{\pi}{12})$ and depict in Figure 14 the frequencies that these regions are used by approximate synthesis of Haar-random operations.

Remark 7.8. In the limit where $\mathcal{S}$ contains all XX interactions, the most efficient circuit for

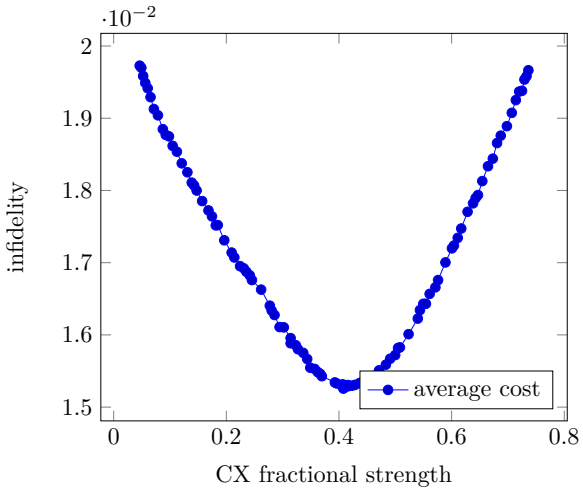


Figure 13: The expected infidelity of a Haar-random operator decomposed approximately into $\mathcal{S}_x = \{CX, XX_x\}$. Uses an additive affine error model with offset $b \approx 1.909 \times 10^{-3}$ and slope $\frac{\pi}{4} \cdot m \approx 5.76 \times 10^{-3}$.

$CAN(a_1, a_2, a_3)$ is given by a product

$$CAN(a_1, a_2, a_3) =$$

$$CAN(a_1) \cdot CAN(0, a_2) \cdot CAN(0, 0, a_3),$$

where each factor in the product is a Weyl reflection of a single XX gate of the same parameter, and where factors are dropped when the relevant parameter vanishes. Under the assumption of an additive affine error model, this establishes a lower bound for how efficient we can expect our circuits to possibly be, as they are assembled from a more restrictive gate set.

Comparing Example 7.5 and Example 7.6, we observe that there are rapidly diminishing returns to enlarging the native gate set. Specialized to the same error model as in the Examples, the performance lower bound argued above is $(3/2 \cdot \frac{\pi}{4}) \cdot m + 3b$, resulting in the table in Figure 15.

Remark 7.9. For a two-qubit unitary U, its *mirror* is the gate $U \cdot SWAP$. The mirror of a canonical gate $CAN(a_1, a_2, a_3)$ is again canonical, given by the formula

$$\begin{cases} CAN\left(\frac{\pi}{4} + a_3, \frac{\pi}{4} - a_2, \frac{\pi}{4} - a_1\right) & \text{when } a_1 \leq \frac{\pi}{4}, \\ CAN\left(\frac{\pi}{4} - a_3, \frac{\pi}{4} - a_2, a_1 - \frac{\pi}{4}\right) & \text{otherwise.} \end{cases}$$

This formula shows that mirroring interchanges the regions of $\mathfrak{A}_{C_2}$ with the most and least infidelity cost, suggesting that our technique may be particularly fruitful at reducing the cost of mirrorable gates. We summarize the numerical results in Figure 17, and we depict in Figure 16 the relative frequency of different circuit templates when synthesizing up to mirroring. The main points are that these two synthesis strategies are "compatible", in that mirroring can be used in tandem with fractional synthesis to effect a combined decrease in infidelity, and that the optimal choice of

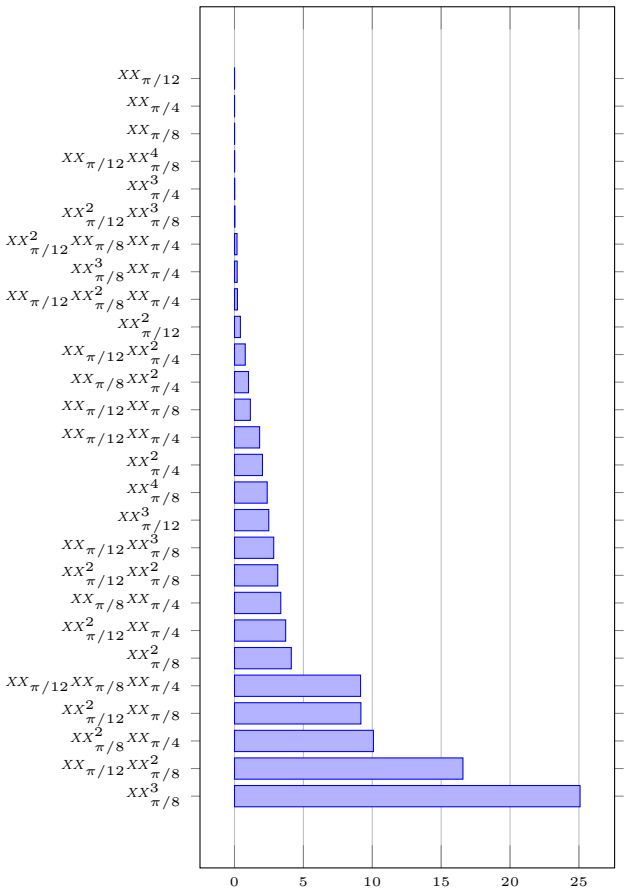


Figure 14: The distribution of circuit types encountered when approximately synthesizing 31,800 Haar-random two-qubit unitaries according to the additive affine error model with offset $b \approx 1.909 \times 10^{-3}$ and slope $\frac{\pi}{4} \cdot m \approx 5.76 \times 10^{-3}$.

Gateset $\mathcal{S}$	Approx.?	argmin $/\frac{\pi}{4}$	min$\langle \mathcal{C}_\mathcal{S} \rangle$	(%)
$\{CX\}$		-	2.279×10^{-2}	(100%)
$\{CX\}$	✓	-	2.058×10^{-2}	(90.3%)
$\{CX, XX_x\}$		0.52041	1.617×10^{-2}	(70.9%)
$\{CX, XX_x\}$	✓	0.51932	1.526×10^{-2}	(67.0%)
$\{CX, XX_x, XX_y\}$		0.61856, 0.41872	1.510×10^{-2}	(66.3%)
$\{CX, XX_x, XX_y\}$	✓	0.66362, 0.43004	1.445×10^{-2}	(63.4%)
$\{XX_{cts}\}$		-	1.437×10^{-2}	(63.1%)
$\{XX_{cts}\}$	✓	-	1.394×10^{-2}	(61.2%)

Figure 15: The expected infidelity of a Haar-random operator with optimal decomposition into optimally chosen gate-sets of various sizes. Uses an additive affine error model with offset $b \approx 1.909 \times 10^{-3}$ and slope $\frac{\pi}{4} \cdot m \approx 5.76 \times 10^{-3}$. Note that $\{XX_{cts}\}$ may be physically unrealistic.

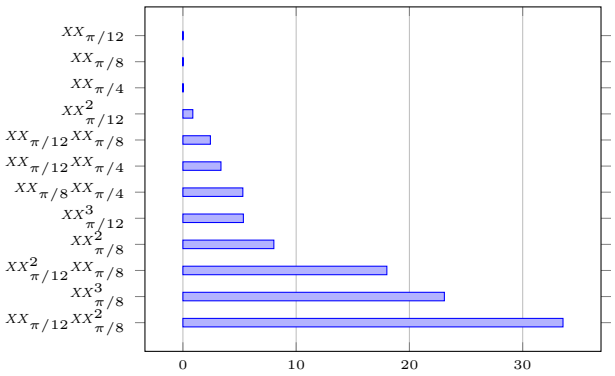


Figure 16: The distribution of circuit types encountered when approximately synthesizing 49,700 Haar-random two-qubit unitaries, allowing mirroring, according to the additive affine error model with offset $b \approx 1.909 \times 10^{-3}$ and slope $\frac{\pi}{4} \cdot m \approx 5.76 \times 10^{-3}$.

finite gateset extension is somewhat different between the two but not wildly so.

Example 7.10. We summarize the statistics on exact, approximate, and mirrored synthesis for the gate sets $\{XX_{\frac{\pi}{4}}\}$, $\{XX_{\frac{\pi}{4}}, XX_{\frac{\pi}{8}}\}$, and $\{XX_{\frac{\pi}{4}}, XX_{\frac{\pi}{8}}, XX_{\frac{\pi}{12}}\}$ in Figure 18. These fractional gates are chosen because they are near to the unconstrained optima and because they are integer fractional iterates of CX, which means they can be easily calibrated and benchmarked. These are to be compared with the precise optima reported in Figure 15 and Figure 17. The main point is that using these fractional gates, rather than the optimal choice of fractional exponents, comes at only a modest cost in infidelity while greatly simplifying engineering.

Acknowledgements

We would like to extend our thanks to David Avis for assisting with `lrs` [21, 22]; to Jay Gambetta, for arranging the productive environment that gave rise to this work; to Isaac Lauer, for keeping us grounded in the realities of working with superconducting hardware; to Douglas Pearson, for helpful comments on a draft of this manuscript; to Michael Mann, for providing the musical score to which this project was carried out; and to the anonymous referees, whose helpful comments significantly improved the readability of this paper.

Gateset $\mathcal{S}$	Approx.?	argmin $/\frac{\pi}{4}$	$\min\langle\mathcal{C}_\mathcal{S}\rangle$	(%)
$\{CX\}$		-	2.279×10^{-2}	(100%)
$\{CX\}$	✓	-	1.895×10^{-2}	(83.2%)
$\{CX, XX_x\}$		0.49970	1.437×10^{-2}	(63.1%)
$\{CX, XX_x\}$	✓	0.46411	1.352×10^{-2}	(59.3%)
$\{CX, XX_x, XX_y\}$		0.49904, 0.24991	1.350×10^{-2}	(59.2%)
$\{XX_{cts}\}$		-	1.304×10^{-2}	(57.2%)
$\{CX, XX_x, XX_y\}$	✓	0.54244, 0.37083	1.300×10^{-2}	(57.0%)
$\{XX_{cts}\}$	✓	-	1.241×10^{-2}	(54.7%)

Figure 17: The expected infidelity of a Haar-random operator *or its mirror* with optimal decomposition into optimally chosen gate-sets of various sizes. Uses an additive affine error model with offset $b \approx 1.909 \times 10^{-3}$ and slope $\frac{\pi}{4} \cdot m \approx 5.76 \times 10^{-3}$. Note that $\{XX_{cts}\}$ may be physically unrealistic.

Gateset $\mathcal{S}$	Approx.?	Mirror?	$\langle\mathcal{C}_\mathcal{S}\rangle$	(%)
$\{CX, XX_{\frac{\pi}{8}}\}$			1.619×10^{-2}	(71.0%)
"	✓		1.534×10^{-2}	(67.3%)
"		✓	1.437×10^{-2}	(63.1%)
"	✓	✓	1.374×10^{-2}	(60.3%)
$\{CX, XX_{\frac{\pi}{8}}, XX_{\frac{\pi}{12}}\}$			1.564×10^{-2}	(68.6%)
"	✓		1.483×10^{-2}	(65.1%)
"		✓	1.352×10^{-2}	(59.3%)
"	✓	✓	1.304×10^{-2}	(57.2%)

Figure 18: The expected infidelity of a Haar-random operator with various optimal synthesis methods into a fixed pair of convenient gate sets. Uses an additive affine error model with offset $b \approx 1.909 \times 10^{-3}$ and slope $\frac{\pi}{4} \cdot m \approx 5.76 \times 10^{-3}$. Compare the expected fidelities with those advertised in Figure 15 and Figure 17.

References

[1] Charles H. Baldwin, Karl Mayer, Natalie C. Brown, Ciarán Ryan-Anderson, and David Hayes. "Re-examining the quantum volume test: Ideal distributions, compiler optimizations, confidence intervals, and scalable resource estimations" (2021). arXiv:2110.14808.

[2] Madan Lal Mehta. "Random matrices". Volume 142 of Pure and Applied Mathematics (Amsterdam), pages xviii+688. Elsevier/Academic Press, Amsterdam. (2004). Third edition.

[3] Qiskit Developers. "Qiskit: An open-source framework for quantum computing" (2021). https://github.com/Qiskit/qiskit-terra.

[4] Eric C. Peterson. "monodromy: Computations in the monodromy polytope for quantum gate sets" (2021). https://github.com/Qiskit/monodromy.

[5] Jun Zhang, Jiri Vala, Shankar Sastry, and K. Birgitta Whaley. "Optimal quantum circuit synthesis from controlled-unitary gates". Phys. Rev. A **69**, 042309 (2004).

[6] Lingling Lao, Prakash Murali, Margaret Martonosi, and Dan Browne. "Designing calibration and expressivity-efficient instruction sets for quantum computing". 2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA) (2021).

[7] Vivek V. Shende, Igor L. Markov, and Stephen S. Bullock. "Minimal universal two-qubit controlled-not-based circuits". Physical Review A**69** (2004).

[8] Andrew W. Cross, Lev S. Bishop, Sarah Sheldon, Paul D. Nation, and Jay M. Gambetta. "Validating quantum computers using randomized model circuits". Phys. Rev. A **100**, 032328 (2019).

[9] Yong-Sheng Zhang, Ming-Yong Ye, and Guang-Can Guo. "Conditions for optimal construction of two-qubit nonlocal gates". Phys. Rev. A **71**, 062331 (2005).

[10] Ming-Yong Ye, Yong-Sheng Zhang, and Guang-Can Guo. "Super controlled gates and controlled gates in two-qubit gate simulations" (2004). arXiv:0407108.

[11] Eric C. Peterson, Gavin E. Crooks, and Robert S. Smith. "Fixed-Depth Two-Qubit Circuits and the Monodromy Polytope". Quantum **4**, 247 (2020).

[12] Nathan Earnest, Caroline Tornow, and Daniel J. Egger. "Pulse-efficient circuit transpilation for quantum applications on cross-resonance-based hardware". Phys. Rev. Research **3**, 043088 (2021).

[13] Petar Jurcevic, Ali Javadi-Abhari, Lev S Bishop, Isaac Lauer, Daniela F Bogorin, Markus Brink, Lauren Capelluto, Oktay Günlük, Toshinari Itoko, Naoki Kanazawa, Abhinav Kandala, George A Keefe, Kevin Krsulich, William Landers, Eric P Lewandowski, Douglas T McClure, Giacomo Nannicini, Adinath Narasgond, Hasan M Nayfeh, Emily Pritchett, Mary Beth Rothwell, Srikanth Srinivasan, Neereja Sundaresan, Cindy Wang, Ken X Wei, Christopher J Wood, Jeng-Bang Yau, Eric J Zhang, Oliver E Dial, Jerry M Chow, and Jay M Gambetta. "Demonstration of quantum volume 64 on a superconducting quantum computing system". Quantum Science and Technology **6**, 025020 (2021).

[14] Cupjin Huang, Dawei Ding, Qi Ye, Feng Wu, Linghang Kong, Fang Zhang, Xiaotong Ni, Yaoyun Shi, Hui-Hai Zhao, and Jianxin Chen. "Towards ultra-high fidelity quantum operations: Sqisw gate as a native two-qubit gate" (2021). arXiv:2105.06074.

[15] B. Kraus and J. I. Cirac. "Optimal creation of entanglement using a two-qubit gate". Physical Review A **63**, 062309 (2001). arXiv:quant-ph/0011050.

[16] Yuriy Makhlin. "Nonlocal properties of two-qubit gates and mixed states, and the optimization of quantum computations". Quantum Information Processing **1**, 243–252 (2002).

[17] Jun Zhang, Jiri Vala, Shankar Sastry, and K. Birgitta Whaley. "Geometric theory of nonlocal two-qubit operations". Phys. Rev. A **67**, 042313 (2003).

[18] Paul Watts, Jiří Vala, Matthias M. Müller, Tommaso Calarco, K. Birgitta Whaley, Daniel M. Reich, Michael H. Goerz, and Christiane P. Koch. "Optimizing for an arbitrary perfect entangler. i. functionals". Phys. Rev. A **91**, 062306 (2015).

[19] Paul Watts, Maurice O'Connor, and Jiří Vala. "Metric structure of the space of two-qubit gates, perfect entanglers and quantum control". Entropy **15**, 1963–1984 (2013).

[20] Marcin Musz, Marek Kuś, and Karol Życzkowski. "Unitary quantum gates, perfect entanglers, and unistochastic maps". Phys. Rev. A **87**, 022111 (2013).

[21] David Avis. "Living with lrs". In Discrete and computational geometry (Tokyo, 1998). Volume 1763 of Lecture Notes in Comput. Sci., pages 47–56. Springer, Berlin (2000).

[22] David Avis and Komei Fukuda. "Reverse search for enumeration". Discrete Appl. Math. **65**, 21–46 (1996).

[23] Lovis Anderson and Benjamin Hiller. "A sweep-plane algorithm for the computation of the volume of a union of polytopes". In Bernard Fortz and Martine Labbé, editors, Operations Research Proceedings 2018. Pages 87–93. Cham (2019). Springer International Publishing.

A Case-work for the approximation theorem

In this Appendix, we chase out the requisite case work to prove Theorem 6.10. We begin with some reductions.

Lemma A.1. *Let $a = (a_1, a_2, a_3)$ be a positive canonical triple, and let $P \subseteq \mathfrak{A}_{C_2}$ be a polyhedron satisfying the reflection-closure property*

$$P = \left\{ \left(\frac{\pi}{2} - b_1, b_2, b_3 \right) \middle| b \in P \right\}.$$

The point $b \in P$ nearest in infidelity distance to the point $a \in \mathfrak{A}_{C_2}$ satisfies $b_1 \leq \frac{\pi}{4}$ if $a_1 \leq \frac{\pi}{4}$ and $b_1 \geq \frac{\pi}{4}$ if $a_1 \geq \frac{\pi}{4}$.

Proof. In the expression

$$\prod_j \cos^2(a_j - b_j) + \prod_j \sin^2(a_j - b_j),$$

the bounds $0 \leq a_2, b_2, a_3, b_3 \leq \frac{\pi}{4}$ entail that the first summand is bounded from below by $\frac{1}{4}\cos^2(a_1 - b_1)$ and the second summand is bounded from above by $\frac{1}{4}\sin^2(a_1 - b_1)$. Notice that replacing b with its reflection $(\frac{\pi}{2} - b_1, b_2, b_3)$ trades the positions in the expression of $\cos^2(a_1 - b_1)$ and $\sin^2(a_1 - b_1)$. Additionally, notice that the expression is maximized when $\cos^2(a_1 - b_1)$ takes on the larger of the two values, i.e., when $\cos^2(a_1 - b_1) \geq \frac{1}{2}$. This then holds exactly when the conclusion of the Lemma does. □

Corollary A.2. *We need only handle the case $a_1 \leq \frac{\pi}{4}$.*

Proof. Theorem 4.1 shows that XX–circuit polytopes are reflection-invariant, so that we may apply the Lemma. Furthermore, since average infidelity is invariant under replacing *both* coordinates by their reflections

$$(a_1, a_2, a_3) \mapsto \left(\frac{\pi}{2} - a_1, a_2, a_3 \right),$$
$$(b_1, b_2, b_3) \mapsto \left(\frac{\pi}{2} - b_1, b_2, b_3 \right).$$

we may reduce to one case of the Lemma, and we choose the case indicated in the Corollary statement. □

From here, we turn to the actual casework, walking first over the codimensions of the various facets of some implicitly understood XX–circuit polytope and then over their possible slopes (noting that in each codimension there is a finite set of possibilities). The codimension 0 and 3 (i.e., top- and bottom-dimensional) cases are trivial:

Lemma A.3 (cf. Remark 6.5)**.** *The infidelity functional $I|_a$ is extremized on the interior of a codimension 0 facet if and only if a is a member of that facet.* □

Lemma A.4. *The codimension 3 facets contribute a finite set of points at which the restricted infidelity function $I|_a$ may be extremized.* □

Lemma A.5. *The infidelity functional $I|_a$ is extremized on the interior of codimension 1 facets coincident with the outer walls of the Weyl alcove if and only if a is a member of that facet.*

Proof. We employ a strategy similar to Remark 4.5: our choice to restriction attention the alcove $\mathfrak{A}_{C_2}$ is artificial, and it is equivalent to optimize the function $\min_{v,w \in W} I(v \cdot a, w \cdot b)$ where W denotes the group of Weyl reflections, where the domain of I is suitably extended by reflection beyond $\mathfrak{A}_{C_2}$, and where b^w is constrained to reside in $\bigcup_{w \in W} w \cdot P$, the closure of P under Weyl reflections. This closure is again a (possibly non-convex, possibly disconnected) polyhedron, but now the points $b \in P$ incident on the outer alcove walls belong to the interior of a codimension 0 facet of $\bigcup_{w \in W} w \cdot P$. Hence, the optimization condition reduces to that of the codimension 0 facet case. □

Lemma A.6. *The infidelity functional $I|_a$ is extremized on the interior of the codimension 1 facets not coincident with the outer walls of the Weyl alcove exactly at the nearest point in Euclidean distance.*

Proof. Each such facet has an associated Lagrange multipliers problem, which we solve in turn. We use the following abbreviations throughout:

$$\partial_j I := \frac{\partial I|_a}{\partial b_j}, \qquad \delta_j := a_j - b_j.$$

The linear constraints on $a, b \in \mathfrak{A}_{C_2}$ describe the following constraints on δ:

$$\delta_1 \in \left[-\frac{\pi}{2}, \frac{\pi}{2} \right], \quad \delta_2 \in \left[-\frac{\pi}{4}, \frac{\pi}{4} \right], \quad \delta_3 \in \left[-\frac{\pi}{4}, \frac{\pi}{4} \right].$$

Referring to Theorem 4.1, we break into cases based on the normal vector of the facet:

$(0, 0, 1)$**:** The Lagrange multiplier constraints are $\partial_1 I = 0$ and $\partial_2 I = 0$, which amount to the trigonometric conditions

$$(c_{2\delta_2} + c_{2\delta_3}) s_{2\delta_1} = 0, \quad (c_{2\delta_1} + c_{2\delta_3}) s_{2\delta_2} = 0.$$

Taking into account the domain constraints on δ, we see that the first equation's is satisfied either when a and b both represent the identity unitary or when $\delta_1 = 0$. Taking into account the deduced constraint $\delta_1 = 0$, the second clause is then satisfied only when $\delta_2 = 0$. Finally, b_3 is determined by being constrained to the frustrum plane.[20]

[20] A special case of this was previously investigated by Cross et al. [8, Equation B8f] when finding the best approximant using a pair of CX gates.

$(1,1,1)$**:** The Lagrange multiplier constraints are $\partial_1 I = \partial_2 I$ and $\partial_2 I = \partial_3 I$, which amount to the trigonometric conditions

$$(c_{\delta_1-\delta_2} + c_{\delta_1+\delta_2} c_{2\delta_3}) s_{\delta_1-\delta_2} = 0,$$
$$(c_{\delta_2-\delta_3} + c_{\delta_2+\delta_3} c_{2\delta_1}) s_{\delta_2-\delta_3} = 0.$$

These equalities are analyzed similarly to as in the previous case. The first equation is satisfied either when both a and b represent the identity unitary or when $\delta_1 = \delta_2$, and the second equality is similarly dispatched to give $\delta_2 = \delta_3$.

$(-1,1,1)$**:** The Lagrange multiplier constraints are $-\partial_1 I = \partial_2 I$ and $-\partial_1 = \partial_3 I$, which amount to the trigonometric conditions

$$(c_{\delta_1+\delta_2} + c_{\delta_1-\delta_2} c_{2\delta_3}) s_{\delta_1+\delta_2} = 0,$$
$$(c_{\delta_1+\delta_3} + c_{\delta_1-\delta_3} c_{2\delta_2}) s_{\delta_1+\delta_3} = 0.$$

Reasoning identically about the domains, we conclude that $-\delta_1 = \delta_2$ and $-\delta_1 = \delta_3$.

In each case, the critical points are seen to lie at the Euclidean projections onto the relevant planes. □

Lemma A.7. *The infidelity functional $I|_a$ is extremized on the interior of the codimension* 2 *facets* not coincident *with the outer walls of the Weyl alcove exactly at the nearest point in Euclidean distance.*

Proof. Again, we are tasked with solving a family of constrained optimization problems. This time, each nondegenerate pair of inner walls intersect at a line with tangent vector v, and we are looking to solve along the line for the condition $\nabla I|_a \cdot v = 0$. To parameterize the line, we select a vertex $b \in \mathfrak{A}_{C_2}$ on it and set

$$\ell(t) = v \cdot t + b.$$

We break v (i.e., the choice of plane pair) into cases.

$(-1,-1,0)$**:** This tangent vector v arises from the intersection of planes with normal vectors $(0,0,1)$ and $(-1,1,1)$. Expanding $(\nabla I|_{a_1,a_2,a_3} \cdot v)$ yields

$$(c_{\delta_1-\delta_2+2\delta_3} + c_{\delta_1-\delta_2-2\delta_3} + 2c_{\delta_1+\delta_2+2t}) s_{\delta_1+\delta_2+2t} = 0,$$

where t is constrained to

$$b_1 - \frac{\pi}{4} \leq t \leq b_2 - b_3.$$

$(1,-1,0)$**:** This tangent vector v arises from the intersection of planes with normal vectors $(0,0,1)$ and $(1,1,1)$. Expanding $(\nabla I|_{a_1,a_2,a_3} \cdot v)$ yields

$$(c_{\delta_1+\delta_2+2\delta_3} + c_{\delta_1+\delta_2-2\delta_3} + 2c_{\delta_1-\delta_2-2t}) s_{\delta_1-\delta_2-2t} = 0,$$

where t is constrained to

$$\frac{1}{2}(b_2 - b_1) \leq t \leq \min\left\{b_2 - b_3, \frac{\pi}{4} - b_1\right\}.$$

$(0,1,-1)$**:** This tangent vector v arises from the intersection of planes of normal vectors $(1,1,1)$ and $(-1,1,1)$. Expanding $(\nabla I|_{a_1,a_2,a_3} \cdot v)$ yields

$$(c_{2\delta_1+\delta_2+\delta_3} + c_{2\delta_1-\delta_2-\delta_3} + 2c_{\delta_2-\delta_3-2t}) s_{\delta_2-\delta_3-2t} = 0,$$

where t is constrained to

$$\frac{1}{2}(b_3 - b_2) \leq t \leq \min\{b_1 - b_2, b_3\}.$$

In each case, the first clause is not satisfiable on the indicated interval, and the second clause contributes at most only the Euclidean critical point. □

Lemma A.8. *The infidelity functional $I|_a$ is extremized on the interior of the codimension* 2 *facets* coincident *with the outer walls of the Weyl alcove exactly at the nearest point in Euclidean distance.*

Proof. As in Lemma A.7, we intend to split over the slopes of the plane-plane intersections. Two of these cases are familiar: since the outer alcove wall $b_3 \geq 0$ shares a normal with the frustrum inequality of Theorem 4.1, the tangent vectors $(-1,-1,0)$ and $(1,-1,0)$ both reappear, and we have already dispatched them in the proof of Lemma A.7. The frustrum inequality contributes one codimension 2 facet not covered by the above: its intersection with the wall $a_2 \geq a_3$ yields a line with tangent vector $(1,0,0)$, and the associated optimization problem is

$$(c_{2\delta_2} + c_{2\delta_3}) s_{2(\delta_1-t)} = 0.$$

The sine factor contributes the Euclidean critical point, and the cosine factor is independent of t.

The remaining cases correspond to "inner creases" in the Weyl-closed solid $\cup_{w\in W} w \cdot P$, and they are treated quite differently. In each case, the strategy is to show that the facet is irrelevant (i.e., has no critical points) unless the outer alcove inequality is tight for the point a, then to use that tightness to simplify the expression further. Our strategy for showing irrelevance is to show that, when a is not a member of an outer facet, $\nabla I|_a$ has a nonnegative inner product with the inward-facing normal of the codimension 2 facet considered as part of the boundary of the inner codimension 1 facet. Taking this as given, we would learn that the extremum then would always lie on the codimension 1 facet, so that we could avoid considering the codimension 2 facet. In fact, this strategy gives us a bit more: even *without* the assumption that a lies off of the outer wall, continuity would show that this conclusion still holds for extrema, since the assumption is only violated at limit points of open regions.[21] Thus, we can avoid investigating even the

[21] Importantly, we are *not* arguing about critical points but about extrema. Critical points can manifest on a boundary via a sequence of points on the bulk which themselves are merely *approximately* critical points, without exactly being critical points. However, any such critical point cannot yield a more extreme value than the value achieved by the function on a sequence of values in the bulk which are extrema for the functional constrained to planes parallel to the outer facet.

aforementioned simplified expressions, leaving open only the task of exhibiting a positive inner product with the inward-facing normal.

$(-2,1,1)$**:** This tangent vector v arises from the intersection of the inner wall with normal $n_i = (1,1,1)$ and outer wall with normal $n_o = (0,1,-1)$. Assuming $a_2 > a_3$, we would like to show that the following quantity is positive:

$$\nabla I|_a \cdot n_i = (c_{a_2-a_3} + c_{a_2+a_3-2t} c_{2(a_1-b_1+2t)}) s_{a_2-a_3},$$

where $(b_1,0,0)$ lies on the line and t satisfies $0 \leq t \leq \frac{1}{3}b_1$.

$(1,1,-2)$**:** This tangent vector v arises from the intersection of the inner wall with normal $n_i = (1,1,1)$ and outer wall with normal $n_o = (1,-1,0)$. Assuming $a_1 > a_2$, we would like to show that the following quantity is positive:

$$\nabla I|_a \cdot n_i = (c_{a_1-a_2} + c_{2(a_3+2t)} c_{a_1+a_2-2b_1-2t}) s_{a_1-a_2},$$

where $(b_1,b_1,0)$ lies on the line and t satisfies $-\frac{1}{3}b_1 \leq t \leq 0$.

$(-2,-1,-1)$**:** This tangent vector v arises from the intersection of the inner wall with normal $n_i = (-1,1,1)$ and outer wall with normal $n_o = (0,1,-1)$. Assuming $a_2 > a_3$, we would like to show that the following quantity is positive

$$\nabla I|_a \cdot n_i =$$

$$(c_{a_2-a_3} + c_{a_2+a_3-2b_1+2t} c_{2(a_1-b_1+2t)}) s_{a_2-a_3},$$

where $(b_1,0,0)$ lies on the line and t satisfies $\frac{1}{2}b_1 - \frac{\pi}{8} \leq t \leq 0$.

In each case, the domain restrictions cause the arguments to sine and cosine to lie in the positive range. □

B Inclusion-exclusion and incidence degeneracy

In uncovering our main results, it was invaluable to be able to calculate the volume of a nonconvex polytope. Not only did volume calculations play an outsized role in Section 7, they also underlie primitive operations. For instance, while containment of a polytope P within a *convex* polytope Q can be checked on vertices, this is not true of two generic polytopes; instead, assuming that P is of constant dimension, $P \subseteq Q$ if and only if $\mathrm{vol}(P) = \mathrm{vol}(P \cap Q)$. For this reason, we found it imperative to have a robust and efficient method for volume calculation.

The process of volume calculation cleaves into two parts: reducing to the convex case, and computing the volume of convex components. Both steps admit several approaches: for instance, the former can be accomplished by (joint) triangulation, and the latter can be accomplished by determinant methods. However, it is difficult to come by implementations of these techniques which are open-source, permissively licensed, accurate / exact, and which operate in high dimension.[22] In our setting, we can often get away with the following: for the second step, use the (somewhat computationally expensive) ability of a computer algebra system, such as `lrs`, to calculate the volume of a single convex polytope; and for the first step, use a variant of inclusion-exclusion.

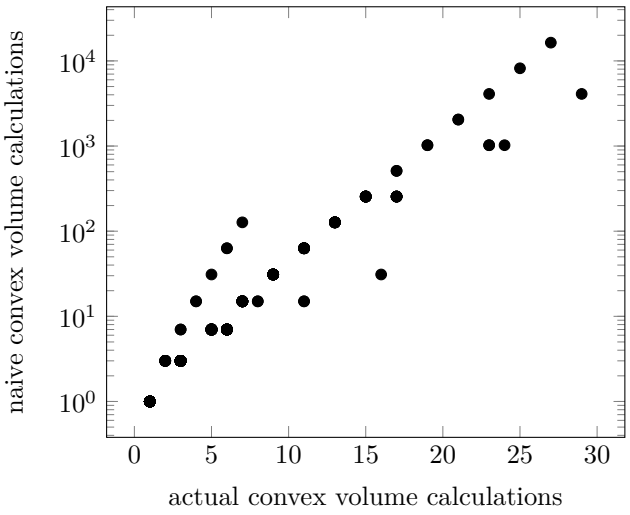


Figure 19: Volume computations made during an example gate set exploration exercise. The vertical coordinate shows the number of convex volume computations required with a naive application of the inclusion-exclusion formula, displayed on a logarithmic scale. The horizontal coordinate shows the number of convex volume computations actually performed when using the method described in Appendix B. Points near the top-left indicate "false complexity" in the convex polytope arrangement, and points near the bottom-right indicate "true complexity".

The naive application of inclusion-exclusion is described by

$$\mathrm{vol}\left(\bigcup_{j\in J} P_j\right) = -\sum_{I\subseteq J} (-1)^{|I|} \mathrm{vol}\left(\bigcap_{i\in I} P_i\right) =: -\sum_{I\subseteq J} (-1)^{|I|} \mathrm{vol}\,(P_I)\,.$$

The terms on the right-hand side are all volumes of convex bodies, hence are individually approachable, but there are $2^{|J|}$ such summands. These summands can be culled in two ways:

1. Terms with vanishing volume are *downward-closed*: If $\mathrm{vol}\, P_I = 0$, then $\mathrm{vol}\, P_{I' \cup I} = 0$ for any I'.
2. Containment is *downward-closed*: If $\mathrm{vol}\, P_I = \mathrm{vol}\, P_{j\cup I}$, then $\mathrm{vol}\, P_{I'\cup I} = \mathrm{vol}\, P_{I'\cup j\cup I}$ for any I'.

[22] See [23] for a notable exception.

For $j \notin I \cup I'$, these pairs of values appear with opposite sign in the larger sum and cancel each other out.

It is simple to cull summands with the first observation: whenever we encounter a summand with vanishing volume, we can skip all of its descendants. The second observation is trickier: after encountering two pairs (j_1, I_1) and (j_2, I_2) which fit the hypothesis, it is possible to double-count a term as belonging to two canceling pairs.

The following procedure accounts for this wrinkle. We will maintain two "skip lists" of indices to ignore:

1. A skip of Type 1 corresponds to an intersection which vanishes exactly, and it is recorded by a single bitmask of the entries which populate I.

2. A skip of Type 2 corresponds to an intersection which cancels with one of its immediate descendants, and it is recorded by a bitmask of the entries which populate I as well as the index j of the descendant (which does not belong to I).

We traverse the possible depths of intersections, and at each depth we traverse the possible intersections at that depth. For each intersection, if it match either skip list, we ignore it and continue to the next intersection at this depth. Otherwise, we compute the volume of this intersection. If the volume vanishes, we add this index to the Type 1 skip list, then continue as if we have done no work at this step. If the volume is equal to one of our immediate predecessors, we add to the Type 2 skip list its index and the extra intersection factor j which witnesses us as its child, then continue as if we have done no work at this step. Otherwise, we add the nonzero contribution to the running alternating sum with the appropriate sign. When we exhaust the possible intersections at this depth, if we have performed no work, we terminate the iteration altogether; otherwise, we proceed to the next depth.

Now, we double back to reintroduce the summands which we previously double-counted, which we formulate in a way to also avoid double-counting the double-countings. Traversing the Type 2 skip list in the order in which it was created, let us consider the t^{th} mask and toggle (I_t, j_t), as well as some intermediate s^{th} mask and toggle (I_s, j_s) with $s < t$ and with $j_t \in I_s$. Double-counting occurs for this pair at an intersection I when the following are met:

1. The t^{th} mask matches $I_t \leq I$.

2. The t^{th} toggle is disabled: $j_t \notin I$.

3. The s^{th} mask matches after the toggle is enabled: $I_s \leq I \cup \{j_t\}$.

4. For all earlier $s' < s$, the s'^{th} mask does not include the t^{th} toggle and additionally does not match I.
5. For all later $s < t' < t$, the t'^{th} mask does not match the toggle-on form $I \cup \{j_t\}$.

Whenever these constraints are met, we reintroduce the summand at I to the running alternating sum. After iterating over all possible values of s and t, the running sum is the true alternating sum.

For any $s < t$, the constraints on I described above are quite strong (and often even contradictory), so that iterating over the possible ways to satisfy these constraints, rather than iterating over I and checking satisfaction, frequently results in loops with few to no iterations. In one instance "in the wild", this strategy reduced a calculation from $2^{14} - 1 \approx 16,000$ convex volume computations to a mere 27 volume computations.

$$
\begin{array}{c:l}
\begin{array}{c}b\text{–coord's unreflected,}\\ a_f = b_3,\\ \text{slant inequality on } a_h\end{array} &
\left\{\begin{array}{rcl}
\alpha_+ + \beta & \geq & b_1 + b_2 + b_3,\\
\alpha_+ - 2\alpha' + \beta & \geq & -b_1 + b_2 + b_3,\\
\alpha_+ - \beta & \geq & -b_1 + b_2 + b_3,\\
\alpha_+ - \alpha' & \geq & b_3,\\
\alpha_+ - \alpha' - \alpha'' + \beta & \geq & b_2.
\end{array}\right.\\
\begin{array}{c}b\text{–coord's reflected,}\\ a_f = b_3,\\ \text{strength inequality on } a_h\end{array} &
\left\{\begin{array}{rcl}
\frac{\pi}{2} + \alpha_+ - 2\alpha' + \beta & \geq & b_1 + b_2 + b_3,\\
-\frac{\pi}{2} + \alpha_+ + \beta & \geq & -b_1 + b_2 + b_3,\\
\alpha_+ - \beta & \geq & b_1 + b_2 + b_3,\\
\alpha_+ - \alpha' & \geq & b_3,\\
\alpha_+ - \alpha' - \alpha'' + \beta & \geq & b_2.
\end{array}\right.\\
\begin{array}{c}b\text{–coord's unreflected,}\\ a_f = b_1,\\ \text{slant inequality on } a_f\end{array} &
\left\{\begin{array}{rcl}
\alpha_+ + -2\alpha' + \beta & \geq & -b_1 + b_2 + b_3,\\
\alpha_+ + \beta & \geq & b_1 + b_2 + b_3,\\
\frac{\pi}{2} - \beta & \geq & b_1 - b_2 + b_3,\\
\alpha_+ + -2\alpha' - \beta & \geq & -b_1 - b_2 + b_3,\\
\alpha_+ - \beta & \geq & b_1 - b_2 + b_3,\\
\alpha_+ - \alpha' - \alpha'' + \beta & \geq & b_3.
\end{array}\right.\\
\begin{array}{c}b\text{–coord's reflected,}\\ a_f = b_1,\\ \text{strength inequality on } a_f\end{array} &
\left\{\begin{array}{rcl}
-\frac{\pi}{2} + \alpha_+ + \beta & \geq & -b_1 + b_2 + b_3,\\
\frac{\pi}{2} + \alpha_+ + -2\alpha' + \beta & \geq & b_1 + b_2 + b_3,\\
\frac{\pi}{2} - \beta & \geq & b_1 - b_2 + b_3,\\
-\frac{\pi}{2} + \alpha_+ - \beta & \geq & -b_1 - b_2 + b_3,\\
\frac{\pi}{2} + \alpha_+ + -2\alpha' - \beta & \geq & b_1 - b_2 + b_3,\\
\alpha_+ - \alpha' - \alpha'' + \beta & \geq & b_3.
\end{array}\right.
\end{array}
$$

Figure 20: Four tables of inequalities describing the four regions of b–coordinates from Remark 5.6. See also Figure 21.

$$
\begin{array}{c}
b\text{–coord's unreflected,} \\
a_f = b_3, \\
\text{slant inequality on } a_h
\end{array}
:
\begin{cases}
-b_3 + \alpha_+ \geq a_h + a_\ell, \\
-b_3 + \alpha_+ - 2\alpha' \geq -a_h + a_\ell, \\
\alpha_+ - \alpha' - \alpha'' \geq a_\ell, \\
b_1 + b_2 + \beta \geq a_h + a_\ell, \\
\pi - b_1 - b_2 - \beta \geq a_h + a_\ell, \\
-b_1 - b_2 + \beta \geq -a_h - a_\ell, \\
b_1 - b_2 + \beta \geq a_h - a_\ell, \\
-b_1 + b_2 + \beta \geq -a_h + a_\ell, \\
b_1 - b_2 - \beta \geq -a_h + a_\ell.
\end{cases}
$$

$$
\begin{array}{c}
b\text{–coord's reflected,} \\
a_f = b_3, \\
\text{strength inequality on } a_h
\end{array}
:
\begin{cases}
\frac{\pi}{2} - b_3 + \alpha_+ - 2\alpha' \geq a_h + a_\ell, \\
-\frac{\pi}{2} - b_3 + \alpha_+ \geq -a_h + a_\ell, \\
\alpha_+ - \alpha' - \alpha'' \geq a_\ell, \\
b_1 + b_2 + \beta \geq a_h + a_\ell, \\
\pi - b_1 - b_2 - \beta \geq a_h + a_\ell, \\
-b_1 - b_2 + \beta \geq -a_h - a_\ell, \\
b_1 - b_2 + \beta \geq a_h - a_\ell, \\
-b_1 + b_2 + \beta \geq -a_h + a_\ell, \\
b_1 - b_2 - \beta \geq -a_h + a_\ell.
\end{cases}
$$

$$
\begin{array}{c}
b\text{–coord's unreflected,} \\
a_f = b_1, \\
\text{slant inequality on } a_f
\end{array}
:
\begin{cases}
-b_1 + \alpha_+ \geq a_h + a_\ell, \\
b_1 + \alpha_+ + -2\alpha' \geq a_h + a_\ell, \\
\alpha_+ - \alpha' - \alpha'' \geq a_\ell, \\
b_2 + b_3 + \beta \geq a_h + a_\ell, \\
\pi - b_2 - b_3 - \beta \geq a_h + a_\ell, \\
-b_2 - b_3 + \beta \geq -a_h - a_\ell, \\
b_2 - b_3 + \beta \geq a_h - a_\ell, \\
-b_2 + b_3 + \beta \geq -a_h + a_\ell, \\
b_2 - b_3 - \beta \geq -a_h + a_\ell.
\end{cases}
$$

$$
\begin{array}{c}
b\text{–coord's reflected,} \\
a_f = b_1, \\
\text{strength inequality on } a_f
\end{array}
:
\begin{cases}
\frac{\pi}{2} - b_1 + \alpha_+ + -2\alpha' \geq a_h + a_\ell, \\
-\frac{\pi}{2} + b_1 + \alpha_+ \geq a_h + a_\ell, \\
\alpha_+ - \alpha' - \alpha'' \geq a_\ell, \\
b_2 + b_3 + \beta \geq a_h + a_\ell, \\
\pi - b_2 - b_3 - \beta \geq a_h + a_\ell, \\
-b_2 - b_3 + \beta \geq -a_h - a_\ell, \\
b_2 - b_3 + \beta \geq a_h - a_\ell, \\
-b_2 + b_3 + \beta \geq -a_h + a_\ell, \\
b_2 - b_3 - \beta \geq -a_h + a_\ell.
\end{cases}
$$

Figure 21: Four tables of inequalities describing the relationship between the a–coordinates and the b–coordinates in the four regions of Remark 5.6. See also Figure 20.