A Complete Equational Presentation of Qudit Circuits

via polycontrolled PROPs

Colin Blake
Université de Lorraine, CNRS, Inria, LORIA

Lisbon, July 2026

FMQC 2026

To be published in MFCS 2026

Qudits are one system with d named states

Definition

A qudit wire represents a d-dimensional space with computational basis

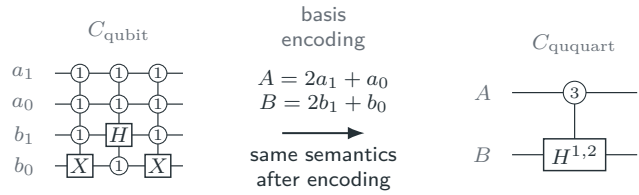
$$|0\rangle, |1\rangle, \ldots, |d-1\rangle.$$

An n-wire circuit represents a unitary on $(\mathbb{C}^d)^{\otimes n}$.

Qudit platforms

- ion levels: trapped-ion qudits ($d \leq 7$) [Ringbauer 22] [Hrmo 23]
- photonic modes: integrated ququart processors [Chi 22]
- transmon levels: qutrit/ququart control [IBM] [Fischer 23]
- bosonic modes: cat-code systems, related but not qudits [Alice & Bob]

Why qudits? More structure per wire



Compression example

Packing two qubits into one ququart turns the branch 11 into the value $A = 3$, and the target pair $01, 10$ into ququart levels $1, 2$.

Circuit-level use cases

wire compression	pack binary data into higher-dimensional wires [Litteken 23]
temporary levels	store intermediate values during Toffoli decompositions [Gokhale 19] [Nikolaeva 23]

One-qudit universality comes from two-level gates

Embedding a 2×2 unitary

$$U = \begin{pmatrix} u_{00} & u_{01} \\ u_{10} & u_{11} \end{pmatrix}.$$

$U^{(i,i+1)}$ is the identity except on adjacent rows and columns $i, i+1$:

$$U^{(i,i+1)} = \begin{bmatrix} I & & & \\ & u_{00} & u_{01} & \\ & u_{10} & u_{11} & \\ & & & I \end{bmatrix}$$

rows and columns ordered as $\cdots, i, i+1, \cdots$

One-qudit synthesis

On the active two-level subspace:

$$P_0(\theta) = \begin{bmatrix} e^{i\theta} & 0 \\ 0 & 1 \end{bmatrix},$$
$$P_1(\theta) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix}.$$

Euler angles write any $U(2)$ operation as a product of

$$P_0(\alpha),\ P_1(\beta),\ H.$$

Adjacent two-level embeddings generate $U(d)$. [Nielsen–Chuang] [Clements 16]

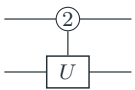
One entangling gate gives n-qudit universality

Entangling control

A value-control is a direct-sum branch.
For $d = 4$,

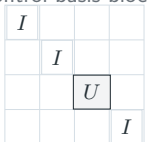
$$\mathrm{ctrl}_2(U) = I \oplus I \oplus U \oplus I.$$

active branch $k = 2$



$\mathrm{ctrl}_2(U)$

control-basis blocks

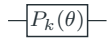


$\mathrm{diag}(I, I, U, I)$

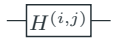
Universal gate set $\mathcal{G}_d$ *[Brylinski–Brylinski 02]*

$$\mathcal{L}_d = \{P_k(\theta),\ H^{(i,j)}\},$$
$$\mathcal{G}_d = \mathcal{L}_d \cup \{\mathrm{ctrl}_1(P_1(\theta))\},$$
$$\langle \mathcal{G}_d \rangle = U(d^n).$$

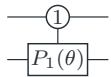
one-qudit



one-qudit



entangling



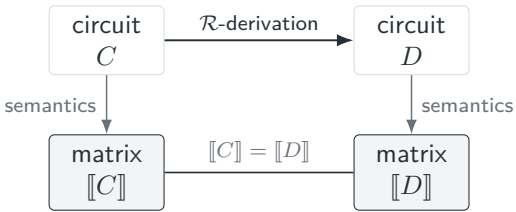
What is a complete presentation?

Definition

A presentation is a gate signature Σ together with equations $\mathcal{R}$, closed under circuit contexts.

$$\mathcal{R} \vdash C = D$$

means that C can be rewritten to D by local rule applications.



Soundness: derivations preserve semantics.
Completeness: equal semantics give a derivation.

Example

Two adjacent H gates cancel; the same local rewrite is allowed inside any circuit context.

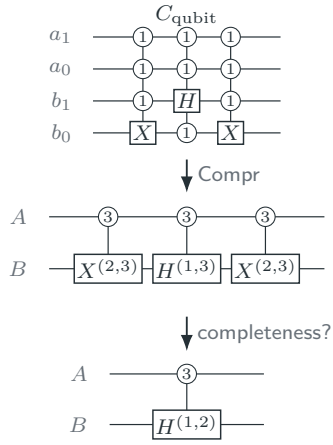
Compressing four basis states into one ququart

Basis encoding

$$\mathbb{C}^2 \otimes \mathbb{C}^2 \overset{\text{enc}}{\cong} \mathbb{C}^4, \quad |a_1 a_0\rangle \mapsto |2a_1 + a_0\rangle$$

Compression of 4 *basis states*

$\mathsf{Compr}(D \circ C) = \mathsf{Compr}(D) \circ \mathsf{Compr}(C)$,
$\mathsf{Compr}(\mathrm{ctrl}_1(\mathrm{ctrl}_1(U))) = \mathrm{ctrl}_3(\mathsf{Compr}(U))$,
$\mathsf{Compr}(\mathrm{ctrl}_1(X)) = X^{(2,3)}$,
$\mathsf{Compr}(\sigma \circ \mathrm{ctrl}_1(H) \circ \sigma) = H^{(1,3)}$.
△ This function here is partially defined.

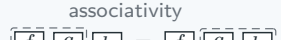


A PRO formalizes circuit composition

Definition

A **PRO** is a strict monoidal category $\mathcal{P}$ with objects $\mathbb{N}$, tensor $m \otimes n = m + n$, and unit 0.

Read n as n **wires**; a morphism $C : n \to m$ as a circuit with n inputs and m outputs.



A PROP makes wire permutations structural

Definition

A **PROP** is a PRO with symmetries $\sigma_{m,n} : m + n \to n + m$.
Diagrammatically, wires may be permuted as structural language.

These are extra **symmetry laws**, on top of the PRO laws from the previous slide.

Unit: $\sigma_{0,n} = \mathrm{id}_n = \sigma_{n,0}$.

$\sigma_{m,n} : m + n \to n + m$

involution

$\sigma_{n,m} \circ \sigma_{m,n} = \mathrm{id}_{m+n}$

naturality

$\sigma \circ (f \otimes \mathrm{id}) = (\mathrm{id} \otimes f) \circ \sigma$

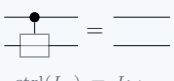
A controlled PROP adds one control constructor

Definition

A **control functor** is an operation $\mathrm{ctrl} : \mathcal{P}_{\mathrm{endo}} \to \mathcal{P}_{\mathrm{endo}}$ with $\mathrm{ctrl}(n) = 1 + n$: it adds one chosen control wire.

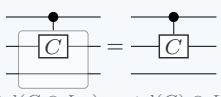
$*$ $\mathcal{P}_{\mathrm{endo}}$: all circuits $n \to n$.

$C : n \to n$
$\mapsto \mathrm{ctrl}(C) : 1 + n \to 1 + n$

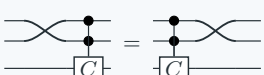


$\mathrm{ctrl}(I_n) = I_{1+n}$

$\mathrm{ctrl}(D \circ C) = \mathrm{ctrl}(D) \circ \mathrm{ctrl}(C)$



$\mathrm{ctrl}(C \otimes I_m) = \mathrm{ctrl}(C) \otimes I_m$



$\mathrm{ctrl}(\mathrm{ctrl}(C)) \circ (\sigma_2 \otimes I)$
$= (\sigma_2 \otimes I) \circ \mathrm{ctrl}(\mathrm{ctrl}(C))$

$\mathrm{ctrl}(\sigma^{-1} \circ C \circ \sigma) = (I \otimes \sigma^{-1}) \circ \mathrm{ctrl}(C) \circ (I \otimes \sigma)$

A polycontrolled PROP adds one constructor per value

Definition

A d-ary **polycontrolled PROP** is a PROP equipped with control functors

$$(\mathrm{ctrl}_k : \mathcal{P}_{\mathrm{endo}} \to \mathcal{P}_{\mathrm{endo}})_{k \in [d]}.$$

Support-sensitivity

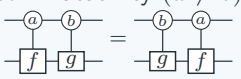
$$\mathrm{supp}(f) \subseteq [d]^n,$$
$$\mathrm{supp}(\mathrm{ctrl}_k(f)) = \{k\} \times \mathrm{supp}(f).$$

Disjoint supports commute.

compatibility

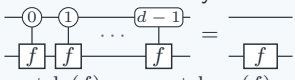
$$\mathrm{ctrl}_a(\mathrm{ctrl}_b(f)) \circ (\sigma_{1,1} \otimes \mathrm{id}_n)$$
$$= (\sigma_{1,1} \otimes \mathrm{id}_n) \circ \mathrm{ctrl}_b(\mathrm{ctrl}_a(f))$$

commutativity $(a \neq b)$



$$\mathrm{ctrl}_a(f) \circ \mathrm{ctrl}_b(g)$$
$$= \mathrm{ctrl}_b(g) \circ \mathrm{ctrl}_a(f)$$

exhaustivity



$$\mathrm{ctrl}_0(f) \circ \cdots \circ \mathrm{ctrl}_{d-1}(f)$$
$$= \mathrm{id}_1 \otimes f$$

Several ways to prove completeness

Normal forms

$$C \iff N(U) \iff D$$
unique

+ canonical representative
+ can become an algorithm
− hard to design for rich syntax
− must prove actual normalization

[Selinger 15] [Li et al. 26]

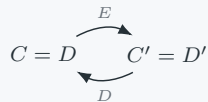
Group presentation

$$\langle G \mid R \rangle \;\cong\; \mathrm{Im}[\![-]\!]$$

+ proves generators and relations
+ good for discrete fragments
− relation sets can explode
− less natural with real parameters

[Bian–Selinger 22] & [23]

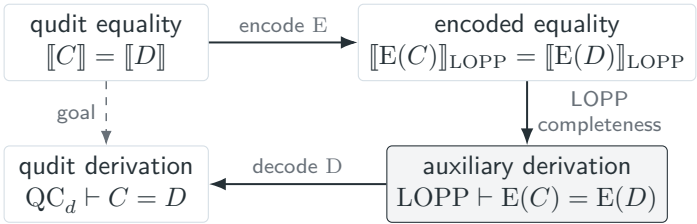
Transfer



+ reuses a complete calculus
+ isolates E, D, and mimicry
− expressivity must match
− auxiliary rules must decode

[Clément et al. 22]

Completeness is transferred by translation



Translation checks

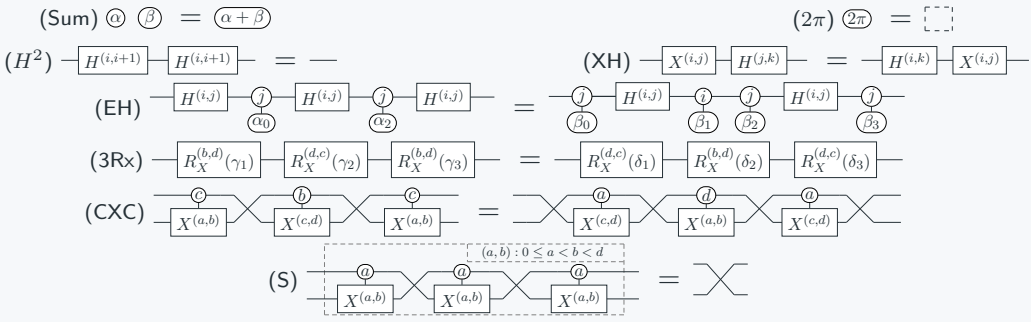
E preserves semantics, $\mathrm{D}(\mathrm{E}(C)) = C$ is derivable, and each LOPP proof step must mimic inside QC_d.

△ LOPP uses $\oplus$, not $\otimes$.

completeness: [Clément et al. 22]

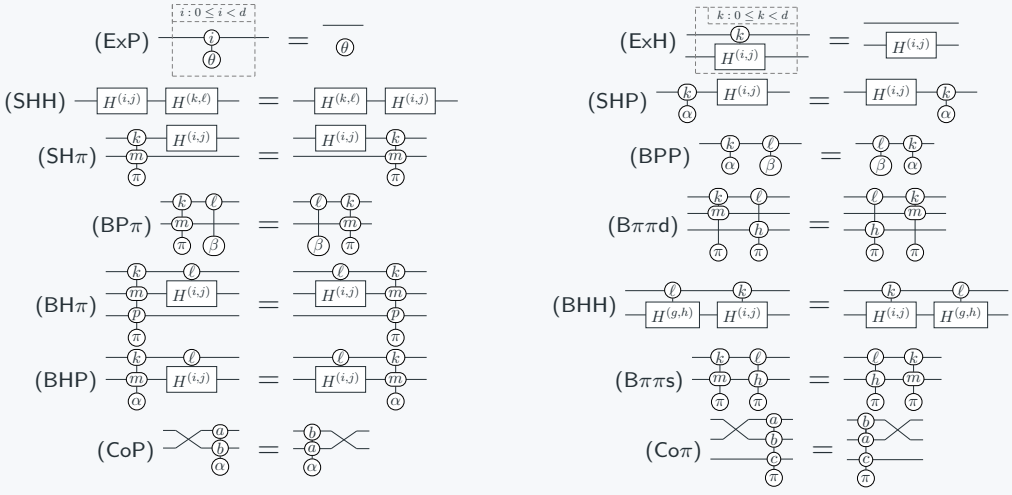
simplified: [Heurtel 25]

Equational theory



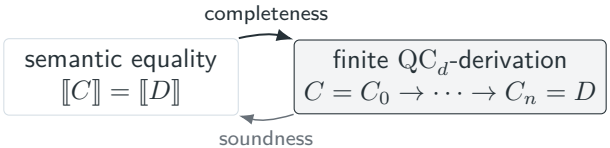
notation: $\mathrm{ctrl}_k(\theta) \equiv \boxed{P_k(\theta)}$

Recovering the properties



Every exact equality has a QC_d-derivation

Theorem. For every finite dimension $d \geq 2$, the finite theory QC_d is **sound and complete** for exact unitary qudit circuits, including global phase.



exact: unitary equality, not approximation or noise

finite: fixed finite schemata for each d

local: bounded circuit windows inside contexts

Implications

- certificates for exact qudit rewrites
- explicit branch syntax
- gate equations separate from control algebra

Future work

- orient fragments as rewrite systems
- native and fault-tolerant gate sets
- arithmetic controls; fragment comparison