arXiv: 1504.03383, 1512.03824, 1605.02756

Factoring with Qutrits

Shawn Xingshan Cui

Stanford University

August 30, 2016

Joint work with Alex Bocharov, Martin Roetteler, Krysta Svore
(QuArC, Microsoft Research)

Why Qutrits?

$$\text{Qubit} = \{|0\rangle, |1\rangle\} \qquad \text{v.s.} \qquad \text{Qutrit} = \{|0\rangle, |1\rangle, |2\rangle\}$$

Why Qutrits?

$$\text{Qubit} = \{|0\rangle, |1\rangle\} \qquad \text{v.s.} \qquad \text{Qutrit} = \{|0\rangle, |1\rangle, |2\rangle\}$$

- Multi-valued logic has computational advantage.

Why Qutrits?

$$\text{Qubit} = \{|0\rangle, |1\rangle\} \qquad \text{v.s.} \qquad \text{Qutrit} = \{|0\rangle, |1\rangle, |2\rangle\}$$

- Multi-valued logic has computational advantage.
- Experimental implementation, e.g. linear ion traps, cold atoms, entangled photons

Why Qutrits?

$$\text{Qubit} = \{|0\rangle, |1\rangle\} \qquad \text{v.s.} \qquad \text{Qutrit} = \{|0\rangle, |1\rangle, |2\rangle\}$$

- Multi-valued logic has computational advantage.
- Experimental implementation, e.g. linear ion traps, cold atoms, entangled photons
- Topological quantum computation (TQC) by non-abelian anyons. Certain anyon system naturally encodes a qutrit.

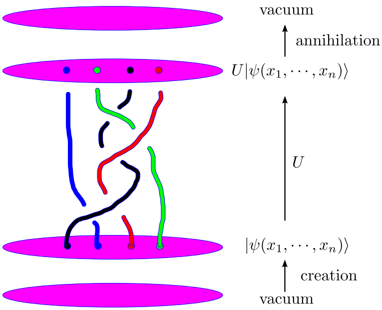
Why Qutrits?

Qubit $= \{|0\rangle, |1\rangle\}$ v.s. Qutrit $= \{|0\rangle, |1\rangle, |2\rangle\}$

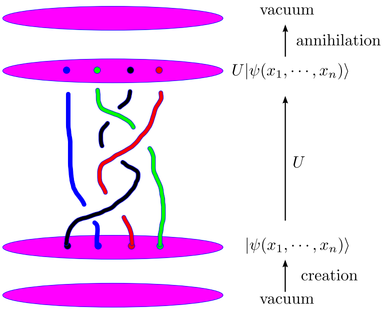
- Multi-valued logic has computational advantage.
- Experimental implementation, e.g. linear ion traps, cold atoms, entangled photons
- Topological quantum computation (TQC) by non-abelian anyons. Certain anyon system naturally encodes a qutrit.

TQC is fault tolerant; has asymptotically better efficiency.

Metaplectic Quantum Computer



Metaplectic Quantum Computer: $SU(2)_4$ anyon system
($\longleftrightarrow$ fractional quantum Hall liquids at $\nu = 8/3$)

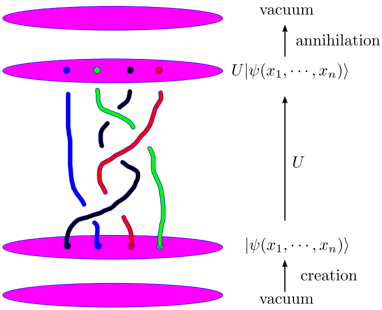


Metaplectic Quantum Computer

Metaplectic Quantum Computer: $SU(2)_4$ anyon system
($\longleftrightarrow$ fractional quantum Hall liquids at $\nu = 8/3$)

Braiding and measurement
$\Rightarrow$ Metaplectic Basis:
Qutrit Clifford + $R_{|2\rangle}$.

- $X|i\rangle = |i+1\rangle$, $Q|i\rangle = \omega^{\delta_{i,2}}|i\rangle$;
  $H|i\rangle = \frac{1}{\sqrt{3}} \sum_{j=0}^{2} \omega^{ij}|j\rangle$;
  $\mathrm{SUM}|i,j\rangle = |i, i+j\rangle$.
- $R_{|2\rangle} = \mathrm{diag}(1, 1, -1)$



Metaplectic Basis

- $R_{|2\rangle}$ is non-Clifford, (not in any Clifford hierarchy).

Metaplectic Basis

- $R_{|2\rangle}$ is non-Clifford, (not in any Clifford hierarchy).
- $R_{|2\rangle}$ is obtained from the magic state $\psi = |0\rangle + |1\rangle - |2\rangle$.

Metaplectic Basis

- $R_{|2\rangle}$ is non-Clifford, (not in any Clifford hierarchy).
- $R_{|2\rangle}$ is obtained from the magic state $\psi = |0\rangle + |1\rangle - |2\rangle$.
- ψ is produced *exactly* by topological measurement in 9/4 trials on average. Much better than any state distillation method.

Metaplectic Basis

- $R_{|2\rangle}$ is non-Clifford, (not in any Clifford hierarchy).
- $R_{|2\rangle}$ is obtained from the magic state $\psi = |0\rangle + |1\rangle - |2\rangle$.
- ψ is produced *exactly* by topological measurement in 9/4 trials on average. Much better than any state distillation method.

Theorem (BCKW, 2015)

Any single qutrit gate can be approximated with precision ϵ by a metaplectic circuit with $R_{|2\rangle}$-count $O(\log(1/\epsilon))$.
(Compared with $O(\log^{3.97}(1/\epsilon))$ Solovay-Kitaev)

Clifford + P_9 Basis

Clifford + P_9, $P_9 = \mathrm{diag}(1, \omega_9, \omega_9^2)$, $\omega_9 = e^{\frac{2\pi i}{9}}$.

Clifford + P_9 Basis

Clifford + P_9, $P_9 = \mathrm{diag}(1, \omega_9, \omega_9^2)$, $\omega_9 = e^{\frac{2\pi i}{9}}$.

P_9:

- Qutrit analog of the qubit $\pi/8$-gate.
- $P_9 \in \mathcal{C}_3$ 3rd Clifford hierarchy.
- Obtained from magic state distillation.
  - magic state $\mu = |0\rangle + \omega_9|1\rangle + \omega_9^2|2\rangle$.
  - distillation complexity: requires $O(\log^3(1/\delta))$ raw magic states to distill one copy of μ with fidelity $1 - \delta$.

Clifford + P_9 Basis

Clifford + P_9, $P_9 = \mathrm{diag}(1, \omega_9, \omega_9^2)$, $\omega_9 = e^{\frac{2\pi i}{9}}$.

P_9:

- Qutrit analog of the qubit $\pi/8$-gate.
- $P_9 \in \mathcal{C}_3$ 3rd Clifford hierarchy.
- Obtained from magic state distillation.
  - magic state $\mu = |0\rangle + \omega_9|1\rangle + \omega_9^2|2\rangle$.
  - distillation complexity: requires $O(\log^3(1/\delta))$ raw magic states to distill one copy of μ with fidelity $1 - \delta$.

Theorem (BCRS, 2015, informal)

One can implement ternary arithmetic (e.g. ternary adder, comparison, multiplication, subtraction, etc.) exactly over Clifford + P_9 basis.

Emulate Qubits in Qutrit Computer

Embed a qubit $\{|0\rangle, |1\rangle\}$ in a qutrit $\{|0\rangle, |1\rangle, |2\rangle\}$.

Emulate Qubits in Qutrit Computer

Embed a qubit $\{|0\rangle, |1\rangle\}$ in a qutrit $\{|0\rangle, |1\rangle, |2\rangle\}$.

Emulate a qubit gate with a qutrit gate. E.g.,

$$\sigma_X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \longrightarrow \quad \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & * \end{pmatrix}$$

Emulate Qubits in Qutrit Computer

Embed a qubit $\{|0\rangle, |1\rangle\}$ in a qutrit $\{|0\rangle, |1\rangle, |2\rangle\}$.

Emulate a qubit gate with a qutrit gate. E.g.,

$$\sigma_X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \longrightarrow \quad \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & * \end{pmatrix}$$

Any qubit algorithm can be emulated in a qutrit computer.

Emulation efficiency?

Emulate Qubits in Qutrit Computer

Embed a qubit $\{|0\rangle, |1\rangle\}$ in a qutrit $\{|0\rangle, |1\rangle, |2\rangle\}$.

Emulate a qubit gate with a qutrit gate. E.g.,

$$\sigma_X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \longrightarrow \quad \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & * \end{pmatrix}$$

Any qubit algorithm can be emulated in a qutrit computer.

Emulation efficiency?

For Shor's Algorithm, compare the cost in terms of raw magic state count.

$$\begin{cases} \text{Clifford} + \pi/8(\text{qubit}) & |0\rangle + e^{\frac{\pi i}{4}}|1\rangle \\ \text{Clifford} + R_{|2\rangle}(\text{qutrit}) & |0\rangle + |1\rangle - |2\rangle \\ \text{Clifford} + P_9(\text{qutrit}) & |0\rangle + e^{\frac{2\pi i}{9}}|1\rangle + e^{\frac{4\pi i}{9}}|2\rangle \end{cases}$$

Shor's Factorization–Period Finding

Quantum part:

Given $a < N$, $(a, N) = 1$, find the smallest number r, such that $a^r = 1 \mod N$.

1. Prepare quantum state proportional to the following superposition:
$$\sum_{k=0}^{N^2} |k\rangle|a^k \mod N\rangle$$
2. Perform quantum Fourier transform of the first register.
3. Measure the first register.

Qutrit Emulation of Shor's Factorization

Proposition (BRS, 2016)

The cost of emulating the binary circuit of period finding is proportional to the cost of emulating the Toffoli gate.

Qutrit Emulation of Shor's Factorization

Proposition (BRS, 2016)

The cost of emulating the binary circuit of period finding is proportional to the cost of emulating the Toffoli gate.

Proposition (BRS, 2016)

The Toffoli gate can be emulated exactly in the Clifford + P_9 basis either

1. *by a four-qutrit circuit with* **6** *P_9 gates (with one ancilla),*
2. *or by a three-qutrit circuit with* **15** *P_9 gates (ancilla free).*

Qutrit Emulation of Shor's Factorization

Proposition (BRS, 2016)

The cost of emulating the binary circuit of period finding is proportional to the cost of emulating the Toffoli gate.

Proposition (BRS, 2016)

The Toffoli gate can be emulated exactly in the Clifford + P_9 basis either

1. *by a four-qutrit circuit with* **6** P_9 *gates (with one ancilla),*
2. *or by a three-qutrit circuit with* **15** P_9 *gates (ancilla free).*

Proposition (BRS, 2016)

The P_9 can be approximated by a metaplectic circuit of $R_{|2\rangle}$-count $6\log_3(1/\epsilon)$.

Comparison of cost of implementing Toffoli

Compare the cost of Toffoli gate in qubit/qutrit models:

	Clean magic states	Raw resources
Clifford $+ \pi/8$	7	$7(2 \log_2(1/\delta))^{2.5}$
Clifford $^A + P_9$	15	$15 \log_2^3(1/\delta)$
Clifford $^B + P_9$	6	$6 \log_2^3(1/\delta)$
Clifford $^A + R_{\|2\rangle}$	15	$90 \log_3(1/\delta)$
Clifford $^B + R_{\|2\rangle}$	6	$36 \log_3(1/\delta)$

Table "Clifford ^{A}" stands for 3-qutrit emulation of the Toffoli gate and "Clifford ^{B}" use 4-qutrit emulation with one clean ancilla prepared with SUM gates.

Comparison of cost of Period finding

Compare the cost of implementing period-finding circuit in qubit/qutrit models:

Circuits	Online width	Offline width
Binary QCLA	$3\,n - w(n)$ (qubits)	$7\,n\,(6\,\log_2(n))^{2.5}$
Clifford ^{A} $+$ P_9	$3\,n - w(n)$ (qutrits)	$15\,n\,(3\,\log_2(n))^3$
Clifford ^{B} $+$ P_9	$4\,n - w(n)$ (qutrits)	$6\,n\,(3\,\log_2(n))^3$
Clifford ^{A} $+$ $R_{\|2\rangle}$	$3\,n - w(n)$ (qutrits)	$90 \times 3\,n\,\log_3(n)$
Clifford ^{B} $+$ $R_{\|2\rangle}$	$4\,n - w(n)$ (qutrits)	$36 \times 3\,n\,\log_3(n)$

Table($w(n)$ is the Hamming weight of n). Clifford ^{A} stands for 3-qutrit emulation of the Toffoli gate and case ^{B} for the 4-qutrit emulation. The last column in metaplectic rows shown the expected average of the probabilistic width.

Summary & Conclusion

Introduced two qutrit basis:

$$\begin{cases} \text{Clifford} + R_{|2\rangle} & \text{Metaplectic TQC} \\ \text{Clifford} + P_9 & \text{Qutrit anolog of qubit Clifford} + \pi/8 \end{cases}$$

Compared the cost of Shor's algorithm in qubit/qutrit models.

Summary & Conclusion

Introduced two qutrit basis:

$$\begin{cases} \text{Clifford} + R_{|2\rangle} & \text{Metaplectic TQC} \\ \text{Clifford} + P_9 & \text{Qutrit anolog of qubit Clifford} + \pi/8 \end{cases}$$

Compared the cost of Shor's algorithm in qubit/qutrit models.

$\Rightarrow$ The solutions over the metaplectic architecture are the most cost-effective in both asymptotic and practical sense. Plus bonus: naturally fault-tolerate.

Summary & Conclusion

Introduced two qutrit basis:

$$\begin{cases} \text{Clifford} + R_{|2\rangle} & \text{Metaplectic TQC} \\ \text{Clifford} + P_9 & \text{Qutrit anolog of qubit Clifford} + \pi/8 \end{cases}$$

Compared the cost of Shor's algorithm in qubit/qutrit models.

$\Rightarrow$ The solutions over the metaplectic architecture are the most cost-effective in both asymptotic and practical sense. Plus bonus: naturally fault-tolerate.

Thank you!