Dichromatic and Trichromatic Calculus for Qutrit Systems

Quanlong Wang    Xiaoning Bian

School of Mathematics and Systems Science, Beihang University, Beijing, China

June 4, 2014
Contents

Some Backgrounds

Qutrit RG Calculus
  Qutrit RG Generators
  Qutrit RG Rules
  Differences Between Qutrit and Qubit Rules
  Euler Decomposition

Qutrit RGB Calculus
  Qutrit RGB Generators
  Qutrit RGB Rules
  RG to RGB Translation

Single Qutrit Quantum Algorithm

Qudit ZX Calculus is Universal
Some Backgrounds

  - Provided a general framework of dagger symmetric monoidal categories for axiomatising complementarity of quantum observables.
  - Introduced the intuitive and universal ZX Calculus for qubits.
  - Introduced a trichromatic graphical calculus.
  - ‘Dichromatic ZX Calculus + Euler angle decomposition of the Hadamard gate = Trichromatic calculus’.
Qutrit RG Generators

We define a category $\mathbf{RG}$ where the objects are $n$-fold monoidal products of an object $\ast$, denoted $\ast^n (n \geq 0)$. In $\mathbf{RG}$, a morphism from $\ast^m$ to $\ast^n$ is a finite undirected open graph from $m$ wires to $n$ wires, built from

\[\delta_Z = \quad \delta_Z^\dagger = \quad \epsilon_Z = \quad \epsilon_Z^\dagger = \quad P_Z(\alpha, \beta) = \frac{\alpha}{\beta}\]

\[\delta_X = \quad \delta_X^\dagger = \quad \epsilon_X = \quad \epsilon_X^\dagger = \quad P_X(\alpha, \beta) = \frac{\alpha}{\beta}\]

\[H = |H| \quad H^\dagger = |H^\dagger|\]

where $\alpha, \beta \in [0, 2\pi)$. For convenience, we denote the frequently used angles $\frac{2\pi}{3}$ and $\frac{4\pi}{3}$ by 1 and 2 respectively.
Qutrit RG Rules

**RG** morphisms are also subject to the following equations:

1. Equations in the following figure.
2. All equations hold under flip of graphs, negation of angles, and exchange of $H$ and $H^\dagger$.
3. All equations hold under flip of colours (except for rules $K2$ and $H2$).
Qutrit RG Rules

\[ \frac{\alpha}{\beta} = \frac{\alpha + \gamma}{\beta + \theta} \]  
\( (S1) \)  
\[ \bullet := \frac{\alpha}{\beta} = \]  
\( (S2) \)

\[ \frac{\alpha}{\beta} = \]  
\( (B1) \)  
\[ \bullet = \]  
\( (B2) \)

\[ \frac{\beta - \alpha}{\alpha - \beta} = \frac{2}{1} = \frac{2}{1} \]  
\( (K1) \)

\[ \frac{\beta - \alpha}{\alpha - \beta} = \frac{2}{1} = \frac{2}{1} \]  
\( (K2) \)

\[ H^\dagger H = H \]  
\( (H1) \)

\[ H^\dagger = H \]  
\( (H2) \)

\[ D := \]  
\( (P1) \)  
\[ \frac{D}{D} = \]  
\( (P2) \)
Some Derived Rules

These equations are very useful when wanting to demonstrate some more complex equalities in describing quantum protocols[4] and algorithms[6].

(1)

(2)

(3)

(4)
Dagger Functor

\[
\begin{align*}
\begin{array}{c}
\left( \alpha \right)^\dagger = \alpha \\
\left( \beta \right)^\dagger = \beta \\
\left( \gamma \right)^\dagger = \gamma \\
\left( H \right)^\dagger = H
\end{array}
\end{align*}
\]

RG is a dagger symmetric monoidal category.
RG Interpretation

We give an interpretation $[\cdot]_{RG} : \text{RG} \rightarrow \text{FdHilb}_Q$

\[
\begin{bmatrix}
\text{green dot}
\end{bmatrix} = |+\rangle \quad \begin{bmatrix}
\text{red dot}
\end{bmatrix} = \langle + | \quad \begin{bmatrix}
\alpha \\
\beta
\end{bmatrix} = |0\rangle \langle 0| + e^{i\alpha} |1\rangle \langle 1| + e^{i\beta} |2\rangle \langle 2|
\]

\[
\begin{bmatrix}
\text{green triangle}
\end{bmatrix} = |0\rangle \langle 0| + |1\rangle \langle 1| + |2\rangle \langle 2| \quad \begin{bmatrix}
\text{red triangle}
\end{bmatrix} = |00\rangle \langle 0| + |11\rangle \langle 1| + |22\rangle \langle 2|
\]

\[
\begin{bmatrix}
\text{red dot}
\end{bmatrix} = |0\rangle \quad \begin{bmatrix}
\text{red circle}
\end{bmatrix} = \langle 0 | \quad \begin{bmatrix}
\alpha \\
\beta
\end{bmatrix} = |+\rangle \langle ++| + e^{i\alpha} |\omega\rangle \langle \omega| + e^{i\beta} |\bar{\omega}\rangle \langle \bar{\omega}|
\]

\[
\begin{bmatrix}
\text{red triangle}
\end{bmatrix} = |+\rangle \langle +++| + |\omega\rangle \langle \omega| + |\bar{\omega}\rangle \langle \bar{\omega}| \quad \begin{bmatrix}
\text{red dot}
\end{bmatrix} = |++\rangle \langle ++| + |\omega\rangle \langle \omega| + |\bar{\omega}\rangle \langle \bar{\omega}|
\]

\[
\begin{bmatrix}
H
\end{bmatrix} = |+\rangle \langle 0| + |\omega\rangle \langle 1| + |\bar{\omega}\rangle \langle 2| \quad \begin{bmatrix}
H^\dagger
\end{bmatrix} = |0\rangle \langle ++| + |1\rangle \langle \omega| + |2\rangle \langle \bar{\omega}|
\]

Differences Between Qutrit and Qubit Rules I

- In qubit case, $\pi = \pi$. For qutrit dualiser, $D := \pi = \pi$.

- The qubit dualizer (identical permutation) is an even permutation, while the qutrit dualizer is an odd permutation.

- There is only one odd permutation $\pi$ in qubit case satisfying

\[
\pi \pi = \pi \pi = \pi \pi = \pi \pi = -\alpha
\]

The qutrit dualizer satisfies

\[
D D = D D = D D = D D = \beta
\]

\[
D \alpha = \beta
\]

\[
D \beta = D
\]
Differences Between Qutrit and Qubit Rules II

In qubit case the $K2$ rule still holds when flipping the colours,

$$\pi \alpha = -\alpha \pi,$$

Whereas it doesn’t hold in qutrit case

$$\frac{1}{2} \beta - \alpha - \alpha = \frac{2}{1} \alpha - \beta,$$

$$\frac{1}{2} \alpha \beta = \frac{2}{1} \alpha \beta.$$
Decomposition of the Hadamard Gate

- Euler decomposition of the Hadamard gate: \[ H = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \]

- The Euler decomposition is not unique:
  \[ H = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \]

- Proof:
  \[ H = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \]
In the qubit case, Duncan and Perdrix [5] proved that the Euler decomposition is not derivable from ZX calculus.

The Euler decomposition is not derivable from $\text{RG}$. 

Proof: We define an alternative interpretation functor $[\cdot]_0 : \text{RG} \rightarrow \text{FdHilb}_Q$ exactly as $[\cdot]_{\text{RG}}$ with the following change:

$$[P_X(\alpha, \beta)]_0 = [P_X(0, 0)]_{\text{RG}} \quad [P_Z(\alpha, \beta)]_0 = [P_Z(0, 0)]_{\text{RG}}$$

This functor preserves all the rules, so its image is indeed a valid model of the theory. However we have the following inequality

$$[H]_0 \neq [P_X(\frac{4\pi}{3}, \frac{4\pi}{3})]_0 \circ [P_Z(\frac{4\pi}{3}, \frac{4\pi}{3})]_0 \circ [P_X(\frac{4\pi}{3}, \frac{4\pi}{3})]_0$$

hence the Euler decomposition is not derivable from $\text{RG}$. 
Qutrit RGB Generators

Similarly, we define a category $\text{RGB}$ where morphism from $\ast^m$ to $\ast^n$ is a finite undirected open graph from $m$ wires to $n$ wires, built from

![Diagram of RGB generators](image-url)
Qutrit RGB Rules

RGB morphisms are subject to the following equations

1. Each colour respects the equations (S1) and (S2).
2. All equations hold under flip of graphs and negation of angles.
3. The following quadruples form bialgebras:

4. 

5. 

6. 
Qutrit RGB Rules

7. 

8. 

9. 

10.
Dagger Functor

- Dagger functor (only showing the blue one):

\[
\begin{align*}
  (\quad)^\dagger &= \quad \\
  (\quad)^\dagger &= \quad \\
  (\quad)^\dagger &= \quad \\
  (\quad)^\dagger &= \quad \\
  (\quad)^\dagger &= \quad \\
  (\quad)^\dagger &= \quad \\
  (\quad)^\dagger &= \quad \\
  (\quad)^\dagger &= \quad \\
\end{align*}
\]

- **RGB** is a dagger symmetric monoidal category.
RGB Interpretation

We give an interpretation $\cdot \_\__{RGB} : RGB \rightarrow FdHilb_Q$

$$\begin{align*}
\begin{bmatrix}
\text{1} & \text{2} & \text{3}
\end{bmatrix} = |+\rangle & \begin{bmatrix}
\text{1} & \text{2} & \text{3}
\end{bmatrix} = \langle + | & \begin{bmatrix}
\text{1} & \text{2} & \text{3}
\end{bmatrix} = |00\rangle \langle 0 | + |11\rangle \langle 1 | + |22\rangle \langle 2 | \\
\begin{bmatrix}
\text{1} & \text{2} & \text{3}
\end{bmatrix} = |0\rangle \langle 00| + |1\rangle \langle 11| + |2\rangle \langle 22| & \begin{bmatrix}
\alpha & \beta
\end{bmatrix} = |0\rangle \langle 0 |+ e^{i\alpha} |1\rangle \langle 1 |+ e^{i\beta} |2\rangle \langle 2 | \\
\begin{bmatrix}
\text{1} & \text{2} & \text{3}
\end{bmatrix} = |u\rangle & \begin{bmatrix}
\text{1} & \text{2} & \text{3}
\end{bmatrix} = \langle u | & \begin{bmatrix}
\text{1} & \text{2} & \text{3}
\end{bmatrix} = |++\rangle \langle +++ | \omega \rangle \langle \omega |+\omega \rangle \langle \bar{\omega} | \langle \bar{\omega} | \\
\begin{bmatrix}
\text{1} & \text{2} & \text{3}
\end{bmatrix} = |\omega\rangle \langle +++| + \bar{\omega} \rangle \langle \omega |+\bar{\omega} \rangle \langle \bar{\omega} | & \begin{bmatrix}
\alpha & \beta
\end{bmatrix} = |+\rangle \langle + |+e^{i\alpha} |\omega\rangle \langle \omega |+e^{i\beta} |\bar{\omega}\rangle \langle \bar{\omega} | \\
\begin{bmatrix}
\text{1} & \text{2} & \text{3}
\end{bmatrix} = |0\rangle & \begin{bmatrix}
\text{1} & \text{2} & \text{3}
\end{bmatrix} = \langle 0 | & \begin{bmatrix}
\text{1} & \text{2} & \text{3}
\end{bmatrix} = |uu\rangle \langle u | + |tt\rangle \langle t | + |vv\rangle \langle v |\\
\begin{bmatrix}
\text{1} & \text{2} & \text{3}
\end{bmatrix} = |u\rangle \langle uu| + |t\rangle \langle tt| + |v\rangle \langle vv | & \begin{bmatrix}
\alpha & \beta
\end{bmatrix} = |u\rangle \langle u |+ e^{i\alpha} |t\rangle \langle t |+ e^{i\beta} |v\rangle \langle v | \\
\end{align*}$$
RGB Interpretation

where \( \omega = e^{\frac{2}{3} \pi i} \), \( \bar{\omega} = e^{\frac{4}{3} \pi i} \), and

\[
\begin{align*}
| + \rangle & = |0\rangle + |1\rangle + |2\rangle \\
| \omega \rangle & = |0\rangle + \omega |1\rangle + \bar{\omega} |2\rangle \\
| \bar{\omega} \rangle & = |0\rangle + \bar{\omega} |1\rangle + \omega |2\rangle
\end{align*}
\]

and

\[
\begin{align*}
| u \rangle & = |0\rangle + \bar{\omega} |1\rangle + \bar{\omega} |2\rangle \\
| t \rangle & = |0\rangle + |1\rangle + \omega |2\rangle \\
| v \rangle & = |0\rangle + \omega |1\rangle + |2\rangle
\end{align*}
\]
**RG to RGB Translation**

We have a functor $\mathcal{T} : \text{RG} \to \text{RGB}$

\[
\mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array}
\]

\[
\mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array}
\]

\[
\mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array}
\]

\[
\mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array}
\]

\[
\mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array}
\]

\[
\mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array}
\]

\[
\mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array}
\]

\[
\mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array}
\]

\[
\mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array}
\]

\[
\mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array} \quad \mathcal{T}\left[\begin{array}{c}
\end{array}\right] = \begin{array}{c}
\end{array}
\]
Recently, Gedik[6] introduces a simple algorithm using only a single qutrit to determine the parity of permutations. Like Deutsch’s algorithm, a speed-up relative to corresponding classical algorithms is obtained. The algorithm can be depicted by the dichromatic calculus:

<table>
<thead>
<tr>
<th>$f$</th>
<th>(0)</th>
<th>(1 2)(0 1)</th>
<th>(1 2)(0 2)</th>
<th>(1 2)</th>
<th>(0 1)</th>
<th>(0 2)</th>
</tr>
</thead>
</table>
| $U_f$ | \[ \begin{array}{c}
0 \\
0
\end{array} \] | \[ \begin{array}{c}
\frac{1}{2} \\
\frac{1}{2}
\end{array} \] | \[ \begin{array}{c}
\frac{2}{1} \\
\frac{2}{1}
\end{array} \] | $D$ | $D$ | $D$
| $U_f \ket{w}$ | \[ \begin{array}{c}
\frac{1}{2} \\
\frac{1}{2}
\end{array} \] | \[ \begin{array}{c}
\frac{1}{2} \\
\frac{1}{2}
\end{array} \] | \[ \begin{array}{c}
\frac{2}{1} \\
\frac{2}{1}
\end{array} \] | $\frac{1}{2}$ | $\frac{1}{2}$ | $\frac{2}{1}$
| Parity | Even | | | | | Odd |
The proof [3] that qudit ZX calculus is universal for quantum mechanics is based on the facts that the d-dimensional phase gates $Z_d, X_d$ are sufficient to simulate all single qudit unitary transforms, where

$$Z_d(b_0, b_1..., b_{d-1}) : b_0 |0\rangle + b_1 |1\rangle + ... + b_{d-1} |d-1\rangle \mapsto |d-1\rangle$$

(the $d$ complex coefficients, $b_0, b_1..., b_{d-1}$ are normalized to unity),

$$X_d(\phi) : \begin{cases} |d-1\rangle &\mapsto e^{i\phi} |d-1\rangle \\ |p\rangle &\mapsto |p\rangle \text{ for } p \neq d-1 \end{cases}$$
Qudit ZX Calculus is Universal

It is proved [3] that some $Z_d$ phase gates can be realized by $X$ phase gate $\Lambda_X(\alpha_1, \alpha_2, ..., \alpha_{d-1})$ in the qudit ZX calculus. However, not every $Z_d$ phase gate can be represented by $\Lambda_X(\alpha_1, \alpha_2, ..., \alpha_{d-1})$. In fact, to realize any $Z_d(b_0, b_1..., b_{d-1})$ in this way, we need to find $\alpha_1, \alpha_2, ..., \alpha_{d-1}$ such that

$$c_j b_0 + c_{j-1} b_1 + ... c_0 b_j + c_{d-1} b_{j+1} + ... c_{j+1} b_{d-1} = d \quad (1)$$

$$c_k b_0 + c_{k-1} b_1 + ... c_0 b_k + c_{d-1} b_{k+1} + ... c_{k+1} b_{d-1} = 0, \quad \forall k \neq j \quad (2)$$

where $c_k = 1 + \sum_{l=1}^{d-1} \eta^{r_k(l)} e^{i \alpha_l}, r_k$ permutes the entries 1 (there is one $r_k$ for each $k$), $j = d$. 
Qudit ZX Calculus is Universal

Since $\sum_{k=0}^{d-1} c_k = d$, summing up all the equations in (1) and (2), we have $\sum_{k=0}^{d-1} b_k = 1$. Of course, not every unit complex vector $(b_0, b_1, ..., b_{d-1})$ satisfies $\sum_{k=0}^{d-1} b_k = 1$ or $\sum_{k=0}^{d-1} b_k = e^{i\alpha}$ up to a global phase.

For example, $(b_0, b_1, ..., b_{d-1}) = (0, 1/\sqrt{2}, 1/\sqrt{2}, 0, ..., 0)$, $d > 2$, is such a counterexample.

The above argument means that we need to find a proof of universality of qudit ZX calculus in another way. We solve this problem by the theory of Lie algebra.
Qudit ZX Calculus is Universal

Let

\[ H = \left\{ \left( \begin{array}{cccc} e^{i\alpha_0} & & & \\ & \ddots & & \\ & & e^{i\alpha_d} & \\ \end{array} \right) \left| \alpha_0, \ldots, \alpha_{d-1} \in \mathbb{R} \right. \right\} \]

\[ V = \frac{1}{\sqrt{d}} \sum_{j,k=0}^{d-1} \omega^{jk} |j\rangle \langle k|, \omega = e^{i\frac{2\pi}{d}}, H' = VH V^{-1}. \]

We give an outline of the proof here, the details will be shown in a forthcoming arXiv paper.

First step: Both \( H \) and \( H' \) are closed connected subgroups of the compact Lie group of unitaries \( G = U(d) \).
Qudit ZX Calculus is Universal

Second step: \( H \) and \( H' \) generate a dense subgroup of \( G \).

By [7], it amounts to showing that \( \mathfrak{h} \) and \( \mathfrak{h}' \) generate \( \mathfrak{g} \) as a Lie algebra, where \( \mathfrak{h} = \text{Lie } H, \mathfrak{h}' = \text{Lie } H', \mathfrak{g} = \text{Lie } G \).
Qudit ZX Calculus is Universal

The basis vectors of Lie algebra $\mathfrak{g}$ consist of

\[
\sigma^{(jk)}_x, (0 \leq j < k \leq d - 1), \sigma^{(jk)}_y, (0 \leq j < k \leq d - 1)
\]

\[
\sigma^{(jk)}_z, (j = 0, 1 \leq k \leq d - 1), il_d
\]

where

\[
\sigma^{(jk)}_x = i |j\rangle \langle k| + i |k\rangle \langle j|, \sigma^{(jk)}_y = |j\rangle \langle k| - |k\rangle \langle j|
\]

\[
\sigma^{(jk)}_z = i |j\rangle \langle j| - i |k\rangle \langle k|
\]

The basis vectors of $\mathfrak{h}$ consist of

\[
\sigma^{(0k)}_z, (1 \leq k \leq d - 1), il_d
\]

thus the basis vectors of $\mathfrak{h}'$ consist of

\[
V \sigma^{(0k)}_z V^{-1}, il_d
\]

By direct calculation, we can prove that $\mathfrak{h}$ and $\mathfrak{h}'$ generate $\mathfrak{g}$. 
Qudit ZX Calculus is Universal

Third step: $H$ and $H'$ generate $G$, i.e., $H$ and $V$ generate $G$. Thus up to a global phase $e^{i\alpha}$, $\Lambda_X$ and $\Lambda_Z$ generate $U(d)$. Here we use the following lemma from [7].

**Lemma**: Let $G$ be a compact Lie group. If $H_1, \ldots, H_k$ are closed connected subgroups and they generate a dense group of $G$, then in fact they generate $G$. 
Acknowledgement

- I am grateful to the financial support from RIMS funding (Research Institute for Mathematical Sciences, Kyoto University).
References


Thanks!