State Injection

Injecting into Rotated Surface Codes from the Corner

Ben Criger

October 26, 2017

1 Introduction

I'm interested to replicate (or at least understand) the work of Ying Li in high-fidelity state injection in surface codes. Namely, I'd like to realise it in rotated surface codes, which use fewer data qubits. This document details an attempt to do so, which may or may not pan out. The key detail is that Ying Li puts the magic state on the corner, and so avoids having lots of random stabilisers nearby. I'd like to see if we can pull off the same trick.

2 ZX-Calculus Snippet

We can express state distillation as a type of state transfer or teleportation (which are equivalent in the ZX-calculus):

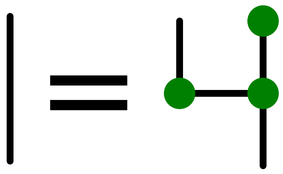


Figure 1: A variation on the state transfer protocol from section 3.3.2 of Coecke/Duncan. We begin with an arbitrary state $|\psi\rangle$ and a $\overline{|+\rangle}$, then merge by projecting into the $+1$-eigenspace of $Z \otimes \bar{Z}$. The resulting state is $\overline{|\psi\rangle}$.

The task now is to find a code for which there's a weight-3 $\bar{Z}$ that goes around a corner. Here's one:

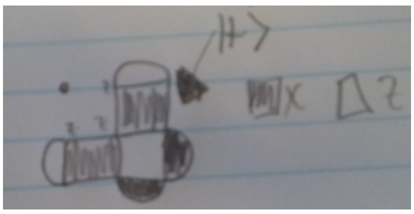


Figure 2: A code with the right $\bar{Z}$ in the right place.

The $\bar{X}$ operators are all weight-two, which lets us break the $\overline{|+\rangle}$ into two Bell pairs and a 4-CAT:

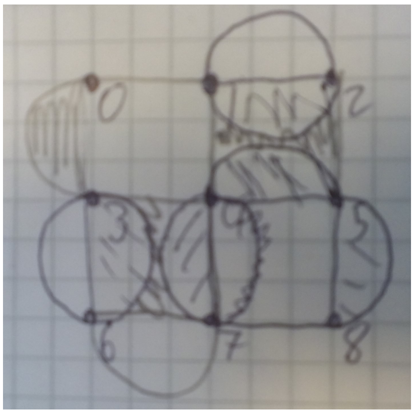


Figure 3: The initial and final stabilisers for the deformation I'm going to try.

In case it's unclear, here's the deformation from initial to final stabilisers/logicals:

$$\begin{aligned} S_{\text{init}} &= \langle X_{12},\, X_{36},\, X_{45},\, X_{47},\, X_{58} \\ &\quad Z_{12},\, Z_{36},\, Z_{4578} \rangle \\ L_{\text{init}} &= \langle X_0,\, Z_0 \rangle \end{aligned} \tag{1}$$

I'm going to label the new stabilisers/logicals by which new operator I measure:

$$\begin{aligned} S_{Z_{0134}} &= \langle X_{1245},\, X_{3467},\, X_{58},\, X_{3456} \\ &\quad Z_{12},\, Z_{36},\, Z_{4578},\, Z_{0134} \rangle \\ L_{Z_{0134}} &= \langle X_{012},\, Z_0 \rangle \end{aligned} \tag{2}$$

$$\begin{aligned} S_{Z_{67}} &= \langle X_{1245},\, X_{3467},\, X_{58} \\ &\quad Z_{12},\, Z_{36},\, Z_{4578},\, Z_{0134},\, Z_{67} \rangle \\ L_{Z_{67}} &= \langle X_{012},\, Z_0 \rangle \end{aligned} \tag{3}$$

$$\begin{aligned} S_{X_{03}} &= \langle X_{1245},\, X_{3467},\, X_{58},\, X_{03} \\ &\quad Z_{12},\, Z_{4578},\, Z_{0134},\, Z_{67} \rangle \\ L_{X_{03}} &= \langle X_{012},\, Z_{036} \rangle \end{aligned} \tag{4}$$

3 Split-Based State Preparation

We can write state preparation of $Z_\alpha|+\rangle$ as a $\bar{Z}$-split based operation:

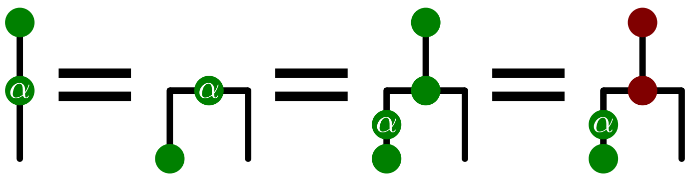
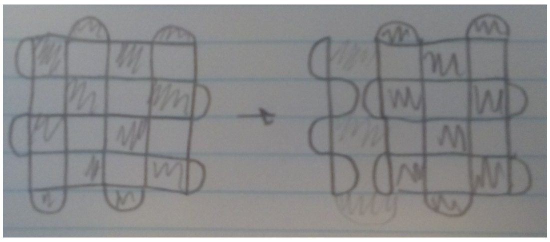


Figure 4: The preparation of $Z_\alpha|+\rangle$ from a single qubit initialised in the $|+\rangle$ or $|0\rangle$ states, by performing the appropriate split (Z or X, respectively), applying the rotation Z_α on one of the split codes and projecting it onto $|+\rangle$.

The repetition code has a weight-one logical, therefore it allows arbitrary transversal rotations about that logical. We can also obtain the repetition code by splitting from a surface code: How to split a repetition code off from a surface code.



After this, we'd have to measure the $\bar{X}$ of the repetition code, which is not ideal, since we can't decode it. However, this may give us a way to prepare magic states that depends only on single-qubit and measurement operations.

4 Open Questions

- Can this be made fault-tolerant?
- What happens if we express state transfer as a bigger merge between multiple codes?
- How about we express a unitary encoder for the surface code in the ZX-calculus and translate it into a series of measurements?
- How do we order and repeat measurements to post-select?