Tensor Network Decoding Beyond 2D

arXiv:2310.10722

Christophe Piveteau, Christopher T. Chubb, and Joesph M. Renes
Institute for Theoretical Physics, ETH Zürich, Switzerland

ETH zürich

Introduction

Tensor network decoding has proven tremendously successful and flexible in decoding 2D-local codes. In this work we extend TN decoding into the third-dimension by generalising the plane-sweep contraction scheme of Ref. [3, 4]. As well as allowing us to study 3D codes, this allows us to decode 2D codes with noisy measurements (under both a phenomenological model and circuit-level noise).

Decoding Pauli Errors

Any Pauli error E can be decomposed as $E = D \cdot S \cdot L$, where L is the logical part of the error, S the stabiliser part, and D the 'destabiliser' part. D is determined by the syndrome, and S is irrelevant to decoding, and so decoding just reduces to determining the L part. As such, optimal decoding can be expressed as:

Given a syndrome s, what is the most likely logical part L among all errors which are compatible with s?

The Role of Degeneracy

A natural starting point would be to find the most likely error among all those compatible with the observed syndrome, this is referred to as the *maximum probability decoder*:

$$\text{MPD:}\quad x \mapsto \arg\max_E \{\Pr(E) \,|\, s(E) = x\}$$

This is not optimal however, as multiple stabiliser-equivalent errors can contribute to the probability of a given logical class. As such the optimal decoder, also known as the *maximum likelihood decoder*, optimises over these logical classes:

$$\text{MLD:}\quad x \mapsto \arg\max_E \left\{ \sum_{F\sim E} \Pr(F) \,\middle|\, s(E) = x \right\}$$

Tensor Network Decoding

The class probabilities $\sum_{F\sim E} \Pr(F)$ admit a tensor network representation, and local codes/noise give local TNs. Specifically, 3D codes of 2D codes with noisy measurements give 3D decoding TNs. By developing an approximate 3D contraction scheme we can therefore approximate the ML decoder in these settings.

Previous works relied on a tensor network construction we refer to as the *generator picture*. As well as this we consider a dual construction we term the *detector picture*, which can be more efficiently contracted for some of the code/noise combinations we consider.

References

[1] C. Piveteau, C.T. Chubb, and J.M. Renes, "Tensor Network Decoding Beyond 2D", arXiv:2310.10722.

[2] C. Gidney, "Stim: a fast stabilizer circuit simulator", Quantum 5, 497 (2021), arXiv:2103.02202.

[3] S. Bravyi, M. Suchara, and A. Vargo, "Efficient algorithms for maximum likelihood decoding in the surface code", Physical Review A 90, 032326 (2014), arXiv:1405.4883.

[4] C.T. Chubb, "General tensor network decoding of 2D Pauli codes", arXiv:2101.04125.

[5] J.C. Bridgeman, C.T. Chubb, "Hand-waving and Interpretive Dance," J. Phys. A: Math. Theor. 50 223001 (2017), arXiv:1603.03039.

Generator picture

In the generator picture we construct the class probabilities from the bottom up. We start with a representative Pauli error E, and summing over all syndrome equivalent errors. In other words, in this picture the class probabilities are evaluated as

$$\Pr(\overline{E}) = \sum \{\Pr(ES) \mid S \in \mathcal{S}\}, \qquad \text{(GenPic)}$$

where $\mathcal{S}$ denotes the stabiliser group.

Detector picture

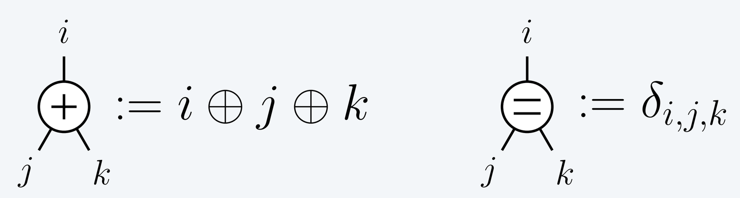
In the detector picture we construct the class probabilities from the top down. Here we start by considering a summation over all Pauli errors, and then imposing the stabiliser and logical measurements to restrict this set down until we arrive at the class probabilities. In other words, in this picture the class probabilities are evaluated as

$$\Pr(\overline{E}) = \sum \left\{ \Pr(F) \,\middle|\, \begin{matrix} s(F) = s(E) \\ l(F) = l(E) \end{matrix} \right\}, \qquad \text{(DetPic)}$$

where s/l denote the outcomes of the syndrome/logical measurements.

Example: 3-qubit repetition code

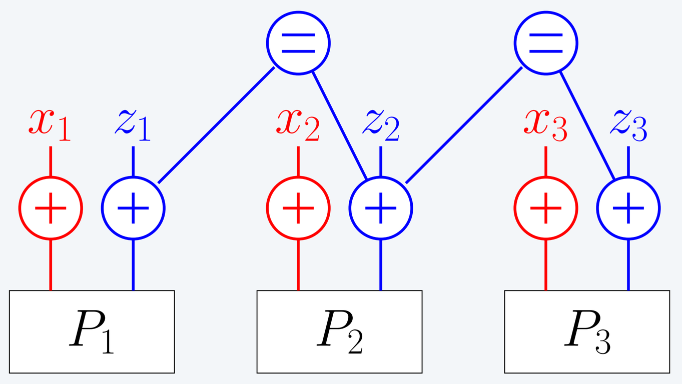
Consider the xor and equality tensors:



In the generator picture we use the stabilisers

$$S_1 = ZZI, \qquad S_2 = IZZ. \qquad (1)$$

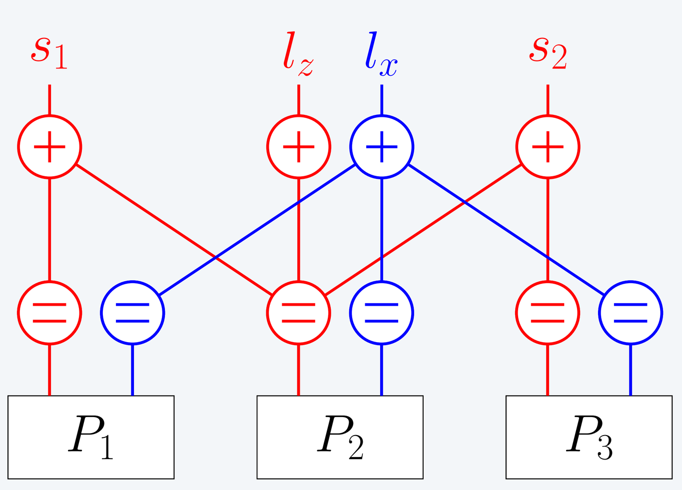
If we consider an error E described by the strings $\vec{x}$ and $\vec{z}$, then the generator form of $\Pr(\overline{E})$ is



In the detector picture we use the detectors,

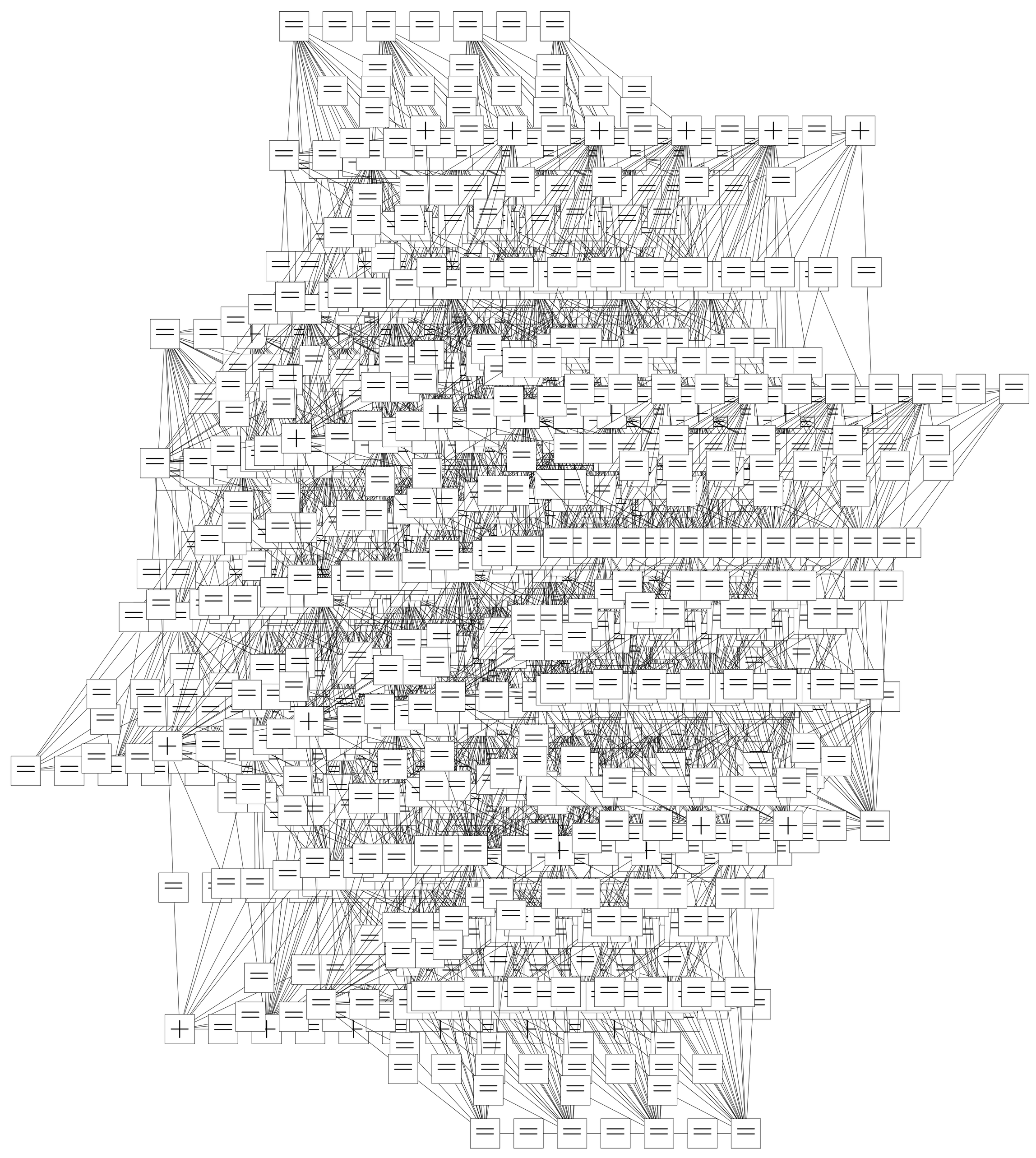
$$\begin{aligned} D_1 &= XII, & L_X &= XXX, \\ D_2 &= IIX, & L_Z &= IZI. \end{aligned} \qquad (2)$$

If we consider an error E whose measurement outcomes are given by s_1, s_2, l_z, l_x, then the detector form of $\Pr(\overline{E})$ is



Circuit-level noise

By representing circuit-level noise through the *detector error model* [2] we can decode it in the detector picture. Doing so gives a very large network however, which prevents us from going to large distances. For example, here is the network for $d = 3$ circuit-level depolarising noise:



Results

	TND	non-TND	Optimal
Point sec.	$3.136^{+0.012}_{-0.014}\%$	$2.93 \pm 0.02\%$	$\approx 3.3\%$
Loop sec.	$22.788^{+0.123}_{-0.107}\%$	$21.55 \pm 0.01\%$	$23.180 \pm 0.004\%$
Dep.	$7.067^{+0.034}_{-0.033}\%$	$6.715 \pm 0.012\%$	?
Circuit dep.	$\approx 0.8\%$	$\approx 0.78\%$	?

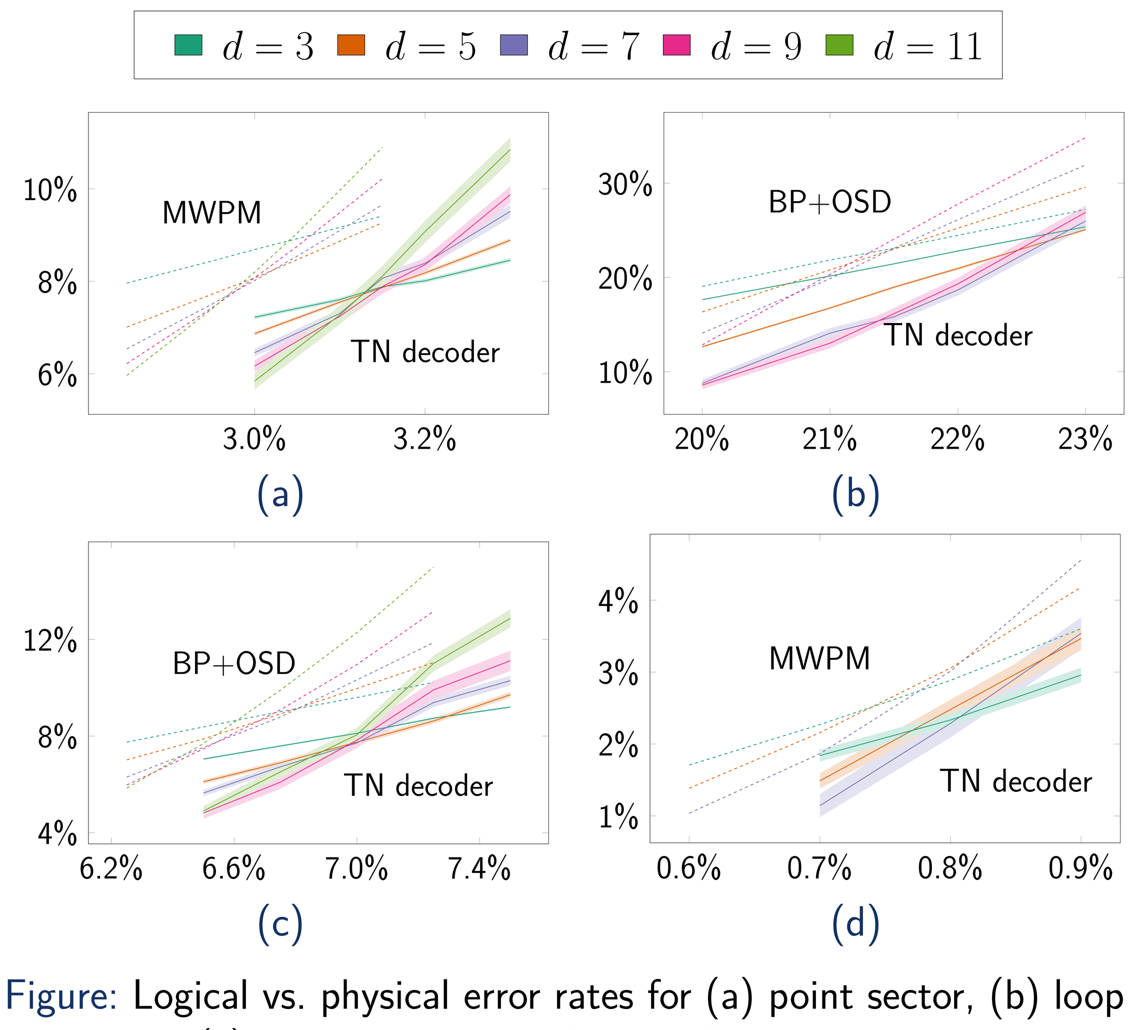
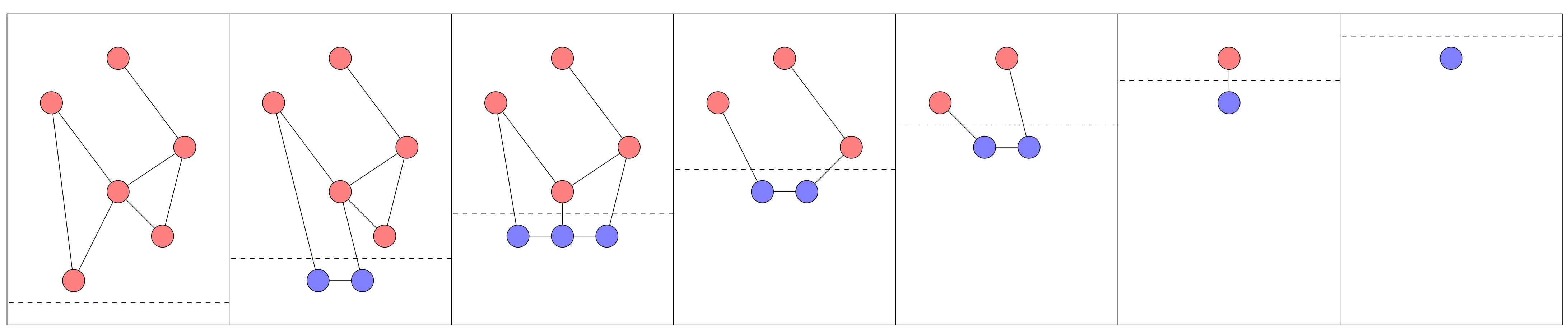


Figure: Logical vs. physical error rates for (a) point sector, (b) loop sector, and (c) depolarizing noise for the 3D unrotated surface code, and circuit level depolarising noise for the rotated 2D surface code.

Plane-sweep Contraction



Contraction of a local tensor network can be approximated by a plane-sweep algorithm. The idea is to contract the network, tensor-by-tensor, into an MPS (2D) or PEPS (3D), using bond truncation to compress down this representation. Refs. [3, 4] developed this method in 2D, and in this work we extend this to 3D.