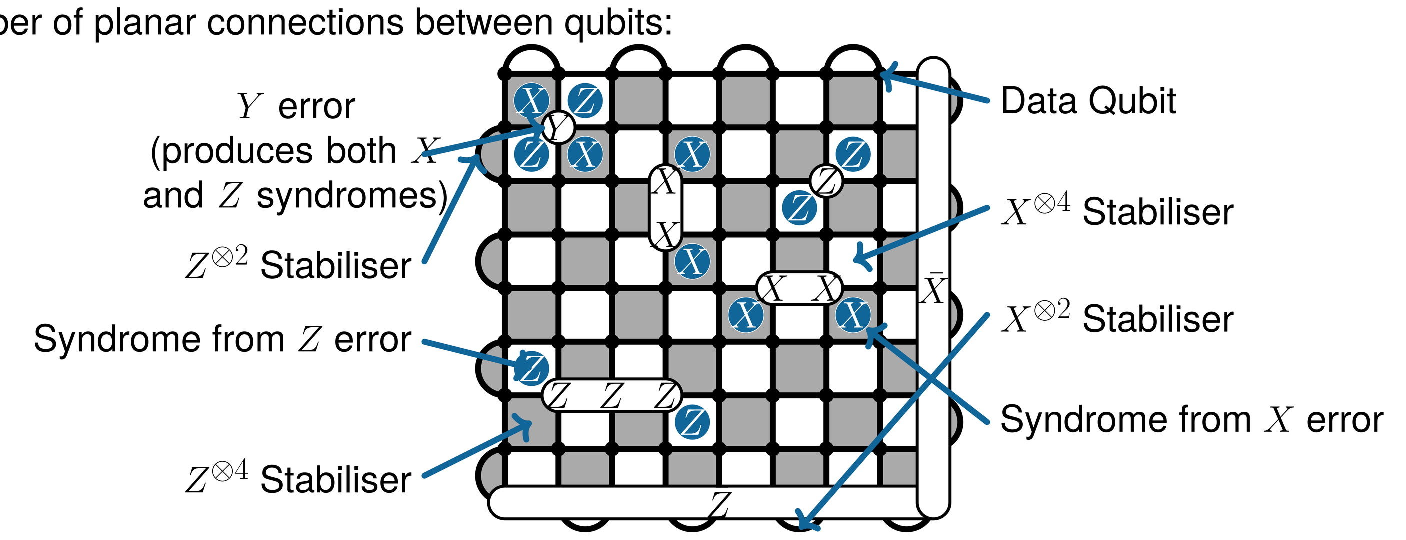
# Multipath Summation for Decoding 2D Topological Codes

QuTech, TU Delft

**Ben Criger** & Imran Ashraf

## The (Rotated) Surface Code

Rotated surface codes are an easy and robust way to store quantum information using a small number of planar connections between qubits:



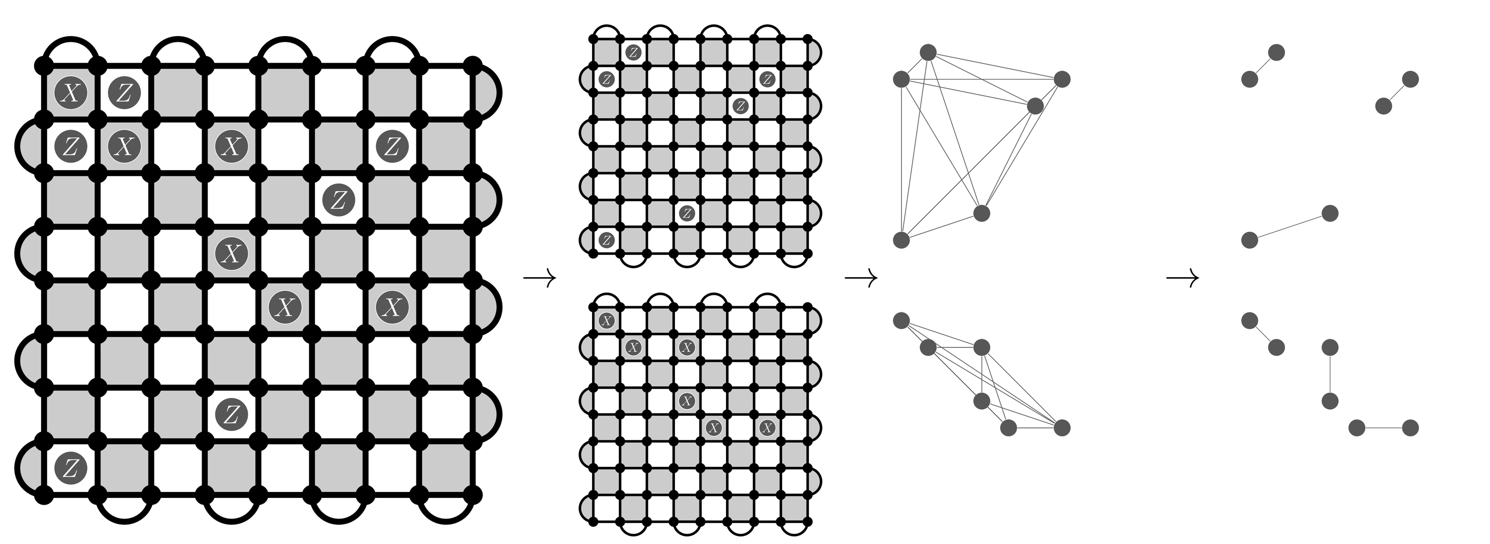

One key property of the code is that non-zero syndromes appear at the endpoints of 1D error chains.

## Decoding by Matching

The likelihood of an error from a model with IID $X/Z$ errors on each data qubit can be expressed as a sum over a graph's edge weights:

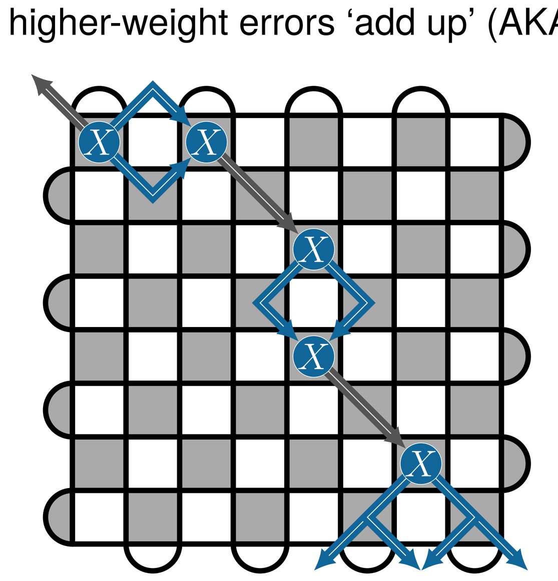
$$\underbrace{p^{|E|}(1-p)^{n-|E|}}_{\text{Likelihood of error}} \xmapsto{\text{decompose}} (1-p)^n \prod_{e\in\text{edges}} \left(\frac{p}{1-p}\right)^{|e|} \xmapsto{\text{remove constant}} \prod_{e\in\text{edges}} \left(\frac{p}{1-p}\right)^{|e|}$$

$$\xmapsto{\text{take log}} \sum_{e\in\text{edges}} \ln\left(\frac{p}{1-p}\right)|e| \propto -\sum_{e\in\text{edges}} |e|$$



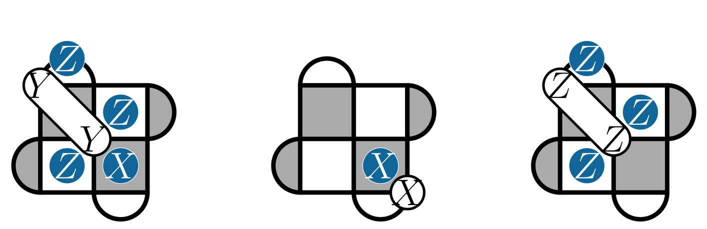
Full Syndrome → Separate $X$ & $Z$ → Complete Graph → Minimum-Weight Perfect Matching

This decoding looks for the single lowest-weight error, so it fails in situations where large numbers of higher-weight errors 'add up' (AKA *degeneracy*):



- Many errors are logically equivalent: $E \cdot E' \in S$
- Many (fewer, but still many) errors can correspond to the same matching
- If $\Omega$ errors correspond to the matching, error likelihood should be increased by a factor of $\Omega$
- If $\frac{p}{1-p} > \frac{1}{\Omega}$, the maximum-likelihood correction changes

Also, the minimum-weight correction for IID $X/Z$ noise isn't necessarily minimum-weight for other models:



Above: An error which is weight $3$ in the IID $X/Z$ model, but weight $2$ in a model where $X$, $Y$ and $Z$ are equally likely (the *depolarizing* error model).

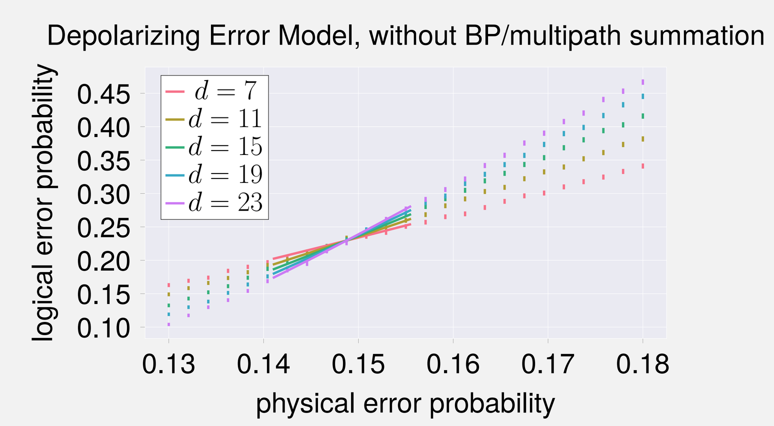
**How well can we adjust minimum-weight perfect matching decoders to ensure that they can correct errors similar to those that occur in experiment, in a way that accounts for as much degeneracy as possible?**

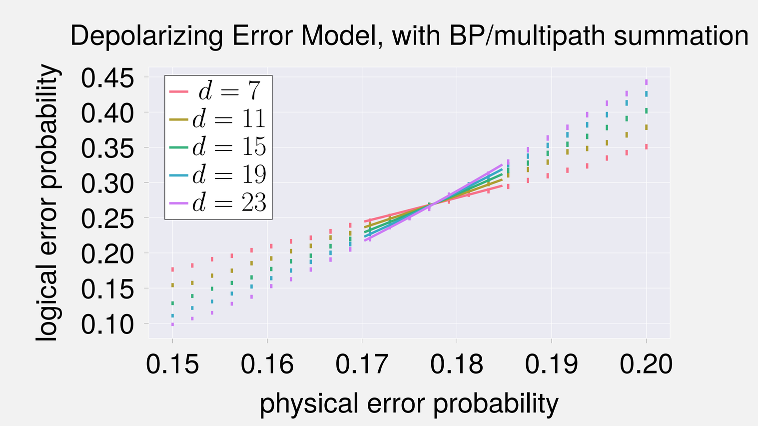
## Acknowledgements

The authors would like to thank Tom O'Brien and Brian Tarasinski for the motivation to study this problem, as well as Barbara Terhal, Kasper Duivenvoorden, Nikolas Breuckmann and Christophe Vuillot for stimulating colloquy.

## Summary

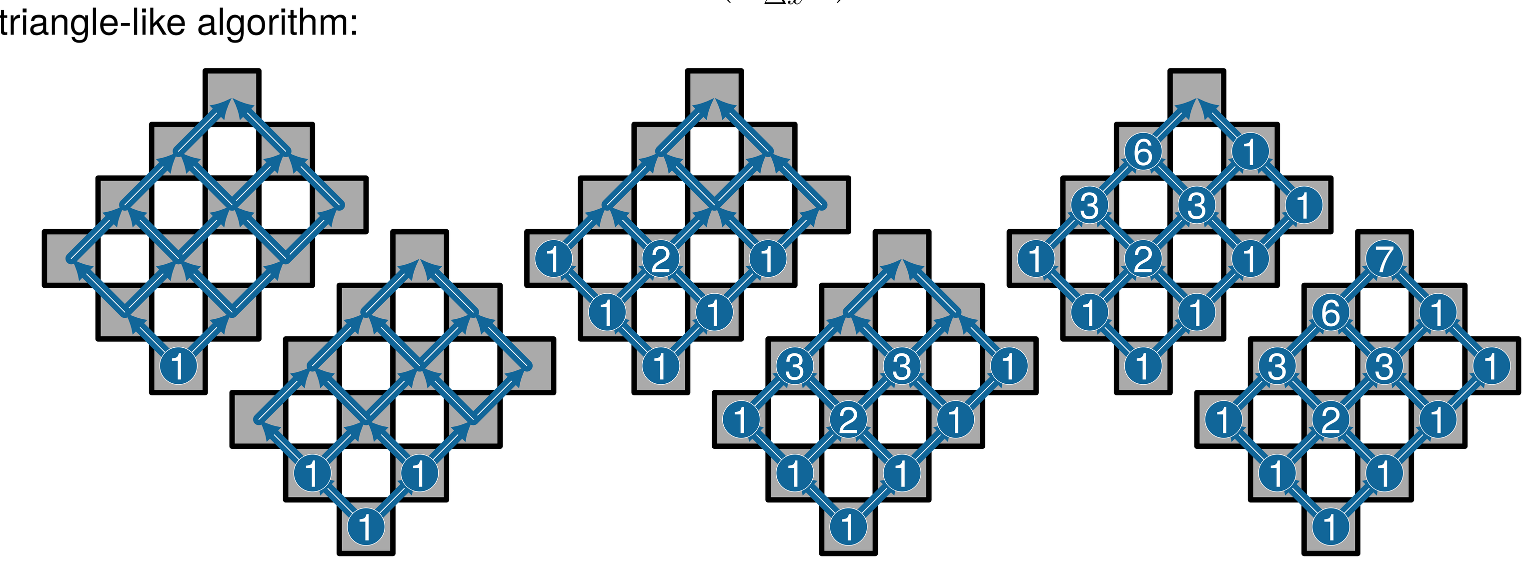
Typical decoders for surface codes are based on minimum-weight perfect matching. They are designed to produce the single likeliest $X$ and $Z$ errors which produce the syndromes observed from $Z$ and $X$ stabilisers, respectively. This leads to sub-optimal threshold error rates, especially for error models with a high probability of $Y$ errors, which produce correlated syndromes. These disadvantages can be mitigated by non-matching-based decoders, though it is not known how to adapt these decoders to repeated measurement, which is required for fault tolerance.
In this work, we improve the performance of matching-based decoders using a new algorithm to produce edge weights corresponding to a larger set of errors, using belief propagation to 'de-correlate' $X$ and $Z$ syndromes. This results in an improvement in the threshold from $14.88\%$ to $17.76\%$.



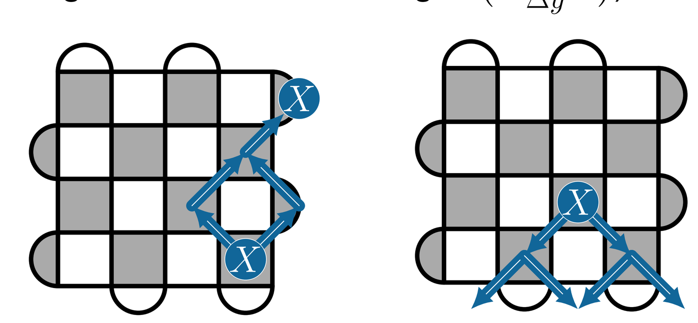




## Path Counting vs. Degeneracy

Barrett and Stace (2010) account for degeneracy in the toric code by noting that the number of errors corresponding to an edge $\Omega_e = \binom{\Delta x + \Delta y}{\Delta x}$, which we can calculate using a Pascal's triangle-like algorithm:

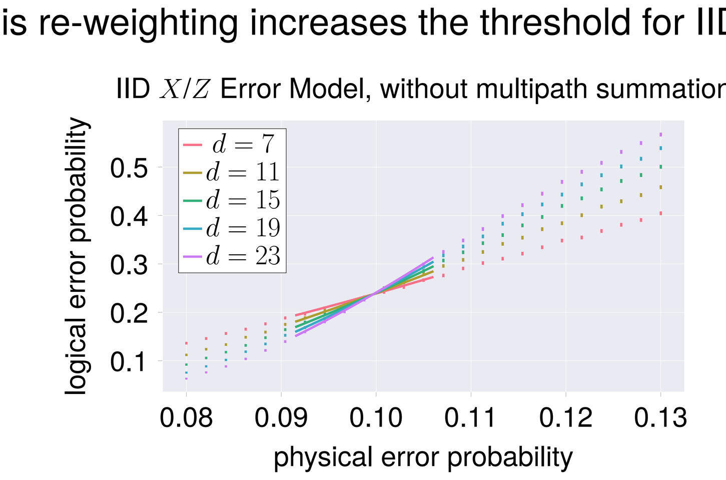


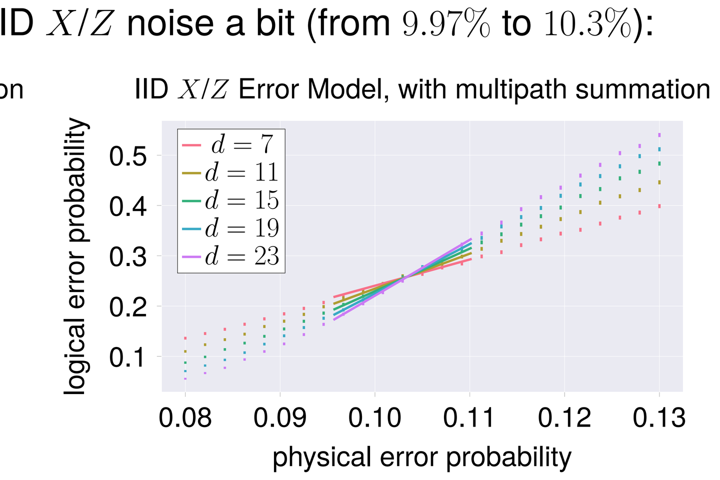
This algorithm also works when there are multiple possible endpoints for a path of errors, or when some points are clipped (though the result is no longer $\binom{\Delta x+\Delta y}{\Delta y}$):



This algorithm works because the number of paths arriving at a vertex is the sum of the number of paths arriving at its *parent vertices*, those with edges 'pointing at' the vertex in question.

This re-weighting increases the threshold for IID $X/Z$ noise a bit (from $9.97\%$ to $10.3\%$):




## Arbitrary Error Probabilities

We can also evaluate the probability that a path exists in the case where each qubit has a different error probability. Given these probabilities, we can express a maximum likelhihood problem as minimum-weight perfect matching, as long as we only use minimum-length paths and we assume that two syndromes are either connected by a path, or there are no errors between them:

$$p_{\text{err}} = \prod_q (1-p_q) \times \prod_{e\in\text{edges}} \sum_{p\in\text{paths}(e)} \prod_{q\in p} o_q \sim \sum_{e\in\text{edges}} \log\left(\sum_{p\in\text{paths}(e)} \prod_{q\in p} o_q\right)$$

## Multi-Path Summation

We can divide $\text{paths}(e)$ into disjoint subsets that pass through each of the parent vertices of the final vertex (or vertices). Each path to the final vertex only traverses a distinct qubit $q_p$, so:

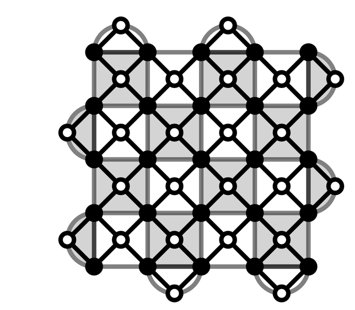
$$\sum_{p\in\text{paths}(e)} \prod_{q\in p} o_q = \sum_{q_p} o_{q_p} \sum_{p'\in\text{paths}(q_p)} \prod_{q\in p'} o_q, \tag{1}$$

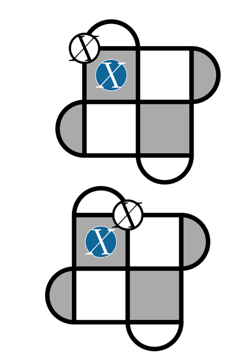
where $\text{paths}(q_p)$ is the set of paths leading from the initial vertex to the predecessor vertex $q_p$.
**This is just like counting paths, but with a weighted sum.**
Now all we need is a way to generate marginal probabilities of error that will be (mostly) accurate.

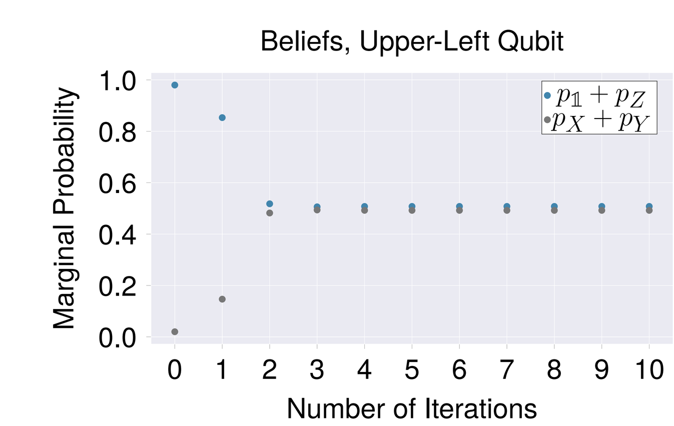
## Belief Propagation vs. $Y$ Errors

Belief proapagation is a message-passing algorithm that calculates marginal probabilities using initial (*prior*) probabilities and observed constraints. It is defined using the *Tanner graph* of the surface code:

- One vertex per qubit, one vertex per check
- Edges connect checks with qubits in their support
- Iterative calculation updates probabilities associated with each qubit vertex
- Guaranteed to work for graphs without cycles (we have lots)





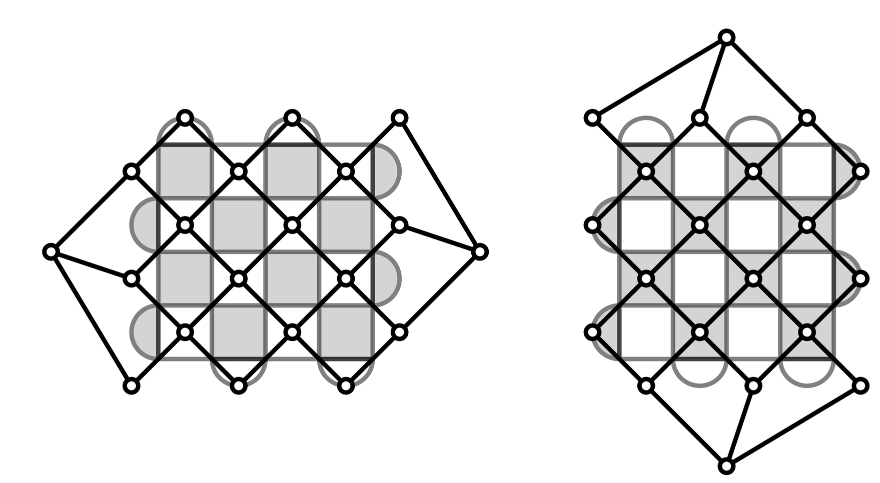



- Belief propagation doesn't always work (if it did, it'd be a decoder)
- If two equally likely errors produce the same syndrome, you can get a *split belief*

Luckily, both qubits with the split belief are considered by multi-path summation, which allows us to produce approximate edge weights for this case (see Summary for results).

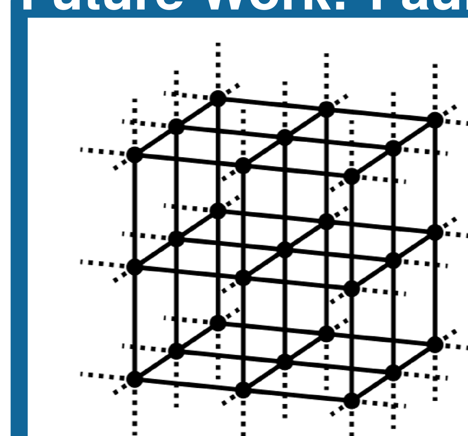
## Future Work: Post-Processing

Bravyi/Suchara/Vargo (2014) produced a decoder which evaluates the probability of a reference correction being wrong. We could evaluate the probability that a correction is wrong, assuming that at most one 'path of differences' exists on the lattice:



We would have to invert $\frac{p}{1-p}$ wherever the test correction has an error.
**Current problem:** we can't use an arbitrarily bad correction here, since we have to be off by at most one logical operator.

## Future Work: Fault Tolerance



Decoding in the presence of measurement/gate noise also reduces to minimum-weight perfect matching, so all the same math ought to apply. Even with the 'diagonal links' introduced by gate errors, the set of minimum-length paths can still be decomposed by parent vertex.
**Current Problem:** I am presenting a poster, and not in Delft with the student who is interested in this research.

## Further Reading

Bravyi, S., Suchara, M., & Vargo, A. (2014). Efficient algorithms for maximum likelihood decoding in the surface code. Physical Review A, 90(3), 032326.
Barrett, S. D., & Stace, T. M. (2010). Fault tolerant quantum computation with very high threshold for loss errors. Physical review letters, 105(20), 200502.
Freund, H., & Grassberger, P. (1989). The ground state of the $+$ or $-J$ spin glass from a heuristic matching algorithm. Journal of Physics A: Mathematical and General, 22(18), 4045.