Extension Implementation Plan

03 March 2025

15:12

The basic version of the algorithm is only for a single round, this can easily be extended to be multiple rounds by making it 3D. You can also then add measurement errors this way.

My current confusion is how to generate the initial erasure model (which is then updated using the syndromes).

I do currently have the probabilities of error for each of the syndromes (these can be found in the detector error model)

There are 3 parts of the extension that need to be implemented:

1. Preparing the input

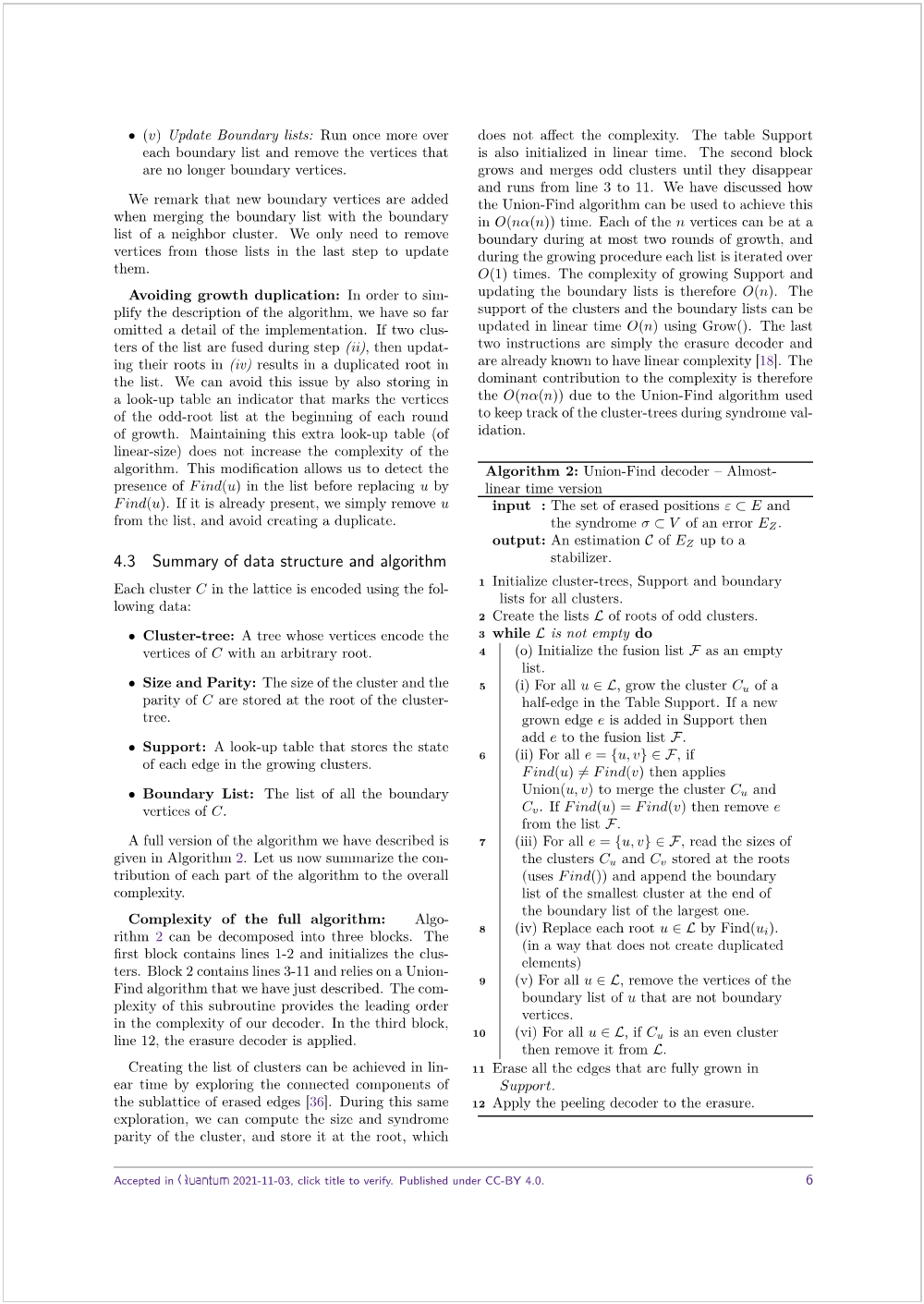
From stim I can easily get:

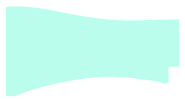
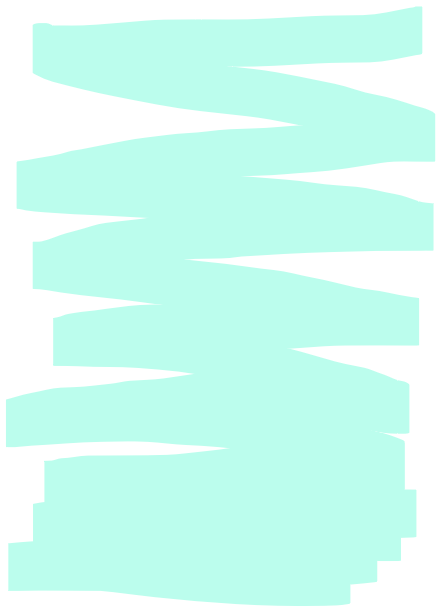
The probabilities of errors for each of the syndrome (as well as the conditional probabilities of the syndrome outputs)

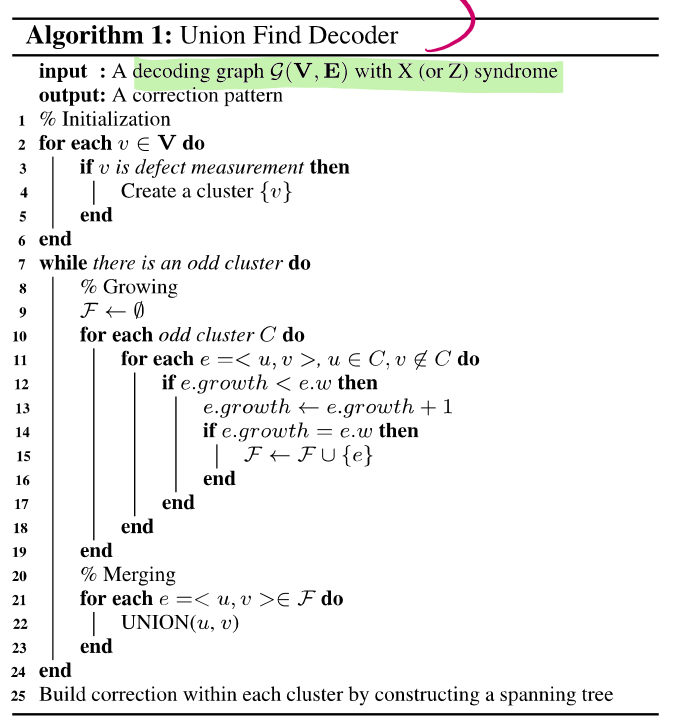
This potentially may be needed to make the initial erasure model, especially if using integer weights proportional to probability

The detection events for each shot (eg each different experiment of the code) - large arrays containing TRUE and FALSE for each of the detectors

1. Implementing the erasure
   1. Syndrome Validation







* 1. Erasure decoder - the peeling algorithm



* 1. Evaluating the decoder

From STIM can get 'observable\_flips' - which I think returns the result of the logical observable (where the logical observables defined by the user).

Basically the output of the erasure decoder will be corrections that need to be done

So if there is a corresponding error and correction event, or no error and no correction event, then it is deemed a success, otherwise it is a failure

