# A QCNN for Quantum State Preparation

## Carnegie Vacation Scholarship

David Amorim

Week 6
(05/08/2024 - 14/08/2024)

## Erratum

The slides for the previous weeks showed the wrong placement of the absolute signs in the definition of SAM. The definition should read:

$$\mathsf{SAM}(|x\rangle, |y\rangle) = 1 - \sum_k |x_k||y_k|. \tag{1}$$

This has now been corrected. Equivalently for WIM.

# Aims for the Week

The following aims were set at the last meeting (05/08/2024):

**Generalise Input States**

When training in superposition, feed in a wider range of input states to ensure the network learns as intended.

**Work on Code and Documentation**

Continue re-structuring and re-documenting the code to ensure a smooth handover.

TRY AND DO SOME BARREN PLATEAU ???

# Table of Contents

# Generalised Input States

- When training in superposition, the QCNN now takes the input state

$$|\psi\rangle_{\text{in}} = \sum_{j=0}^{2^n-1} c_j |j\rangle \tag{2}$$

  where the coefficients $c_j \sim \frac{1}{\sqrt{2^n}}$ are randomly sampled each epoch
- The range of the random sampling is controlled by a hyper-parameter $\delta$, $0 \leq \delta \leq 1$
- For instance, $\delta = 0$ gives $c_j = \frac{1}{\sqrt{2^n}}$ while $\delta = 1$ gives $c_j \in (0, 1)$
- This generalisation should ensure that the network learns the operation $|j\rangle |0\rangle \mapsto |j\rangle |\Psi'(j)\rangle$ as opposed to just learning how to produce a particular fixed state

# Results

Amplitudes after applying $\tilde{Q}$ with $\Psi(f) \sim f^2$ and the input register in initial state $\hat{H}\,|0\rangle$ ($L = 6$, $m = 3$, SAM, 600 epochs):

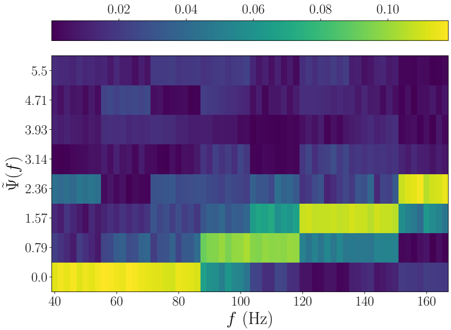


Figure 1: $\delta = 0$

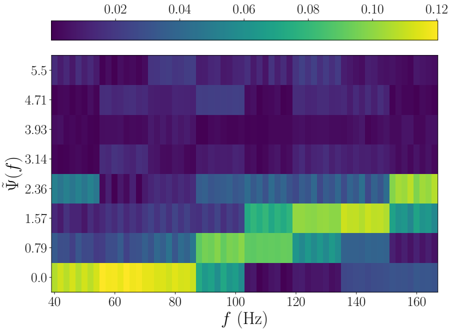


Figure 2: $\delta = 0.2$

# Results

Amplitudes after applying $\tilde{Q}$ with $\Psi(f) \sim f^2$ and the input register in initial state $\hat{H}\,|0\rangle$ ($L = 6$, $m = 3$, SAM, 600 epochs):

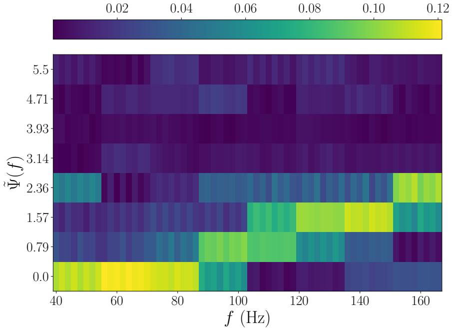


Figure 3: $\delta = 0.4$

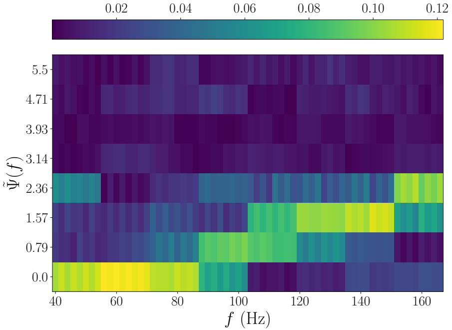


Figure 4: $\delta = 0.6$

# Results

Amplitudes after applying $\tilde{Q}$ with $\Psi(f) \sim f^2$ and the input register in initial state $\hat{H}\,|0\rangle$ ($L = 6$, $m = 3$, SAM, 600 epochs):

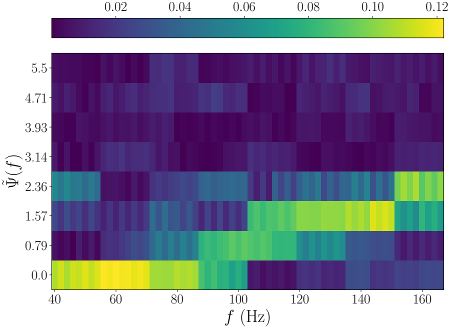


Figure 5: $\delta = 0.8$

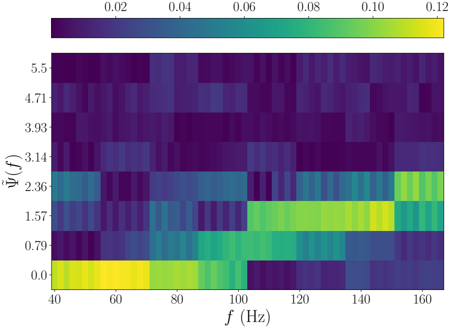


Figure 6: $\delta = 1.0$

# Results

- It seems that randomised input states ($\delta \neq 0$) actually have an adverse effect on performance
- Notably, increasing $\delta$ leads to thin 'stripes' in the amplitude plots, i.e. sudden changes in amplitude between neighbouring states
- This could suggest a structural defect in input layer design

Notably, there is a 'stripey' texture... also, try playing around with control state ... ..

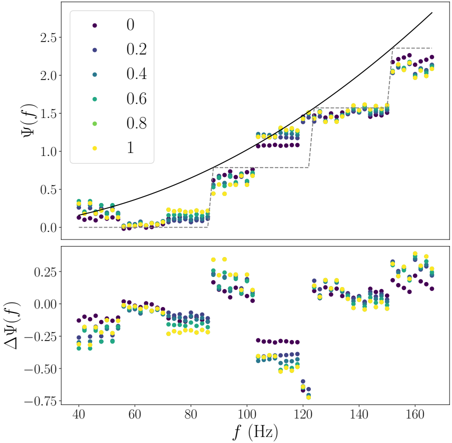


Figure 7: Comparing the effect of $\delta$ values for $\Psi(f) \sim f^2$ ($L = 6$, $m = 3$, SAM, 600 epochs)

# Table of Contents

# Code and Documentation

- Continued work on code documentation: now hosted online here

TO DO:

a. write doc strings for testQNN, trainQNN, ampl_trainQNN
b. make sure to update doc files on github (-o /docs)
c. add plotting into comman-line feature (re-write plotting_tools and check_plots ...)
d. integrate encode.py

# Table of Contents

# Next Steps

- Start work on poster for Carnegie