Implementing generalized measurements with mid-circuit measurements

Gideon Uchehara, Peter Ivashkov, Derek Wang, Alireza Seif

October 2022

Abstract

Generalized measurements can be implemented through a binary search tree with a depth logarithmic in the number of possible outcomes of a positive operator valued measurement (POVM). This protocol is realized by coupling the measured system to a single auxiliary qubit that is iteratively measured and conditionally operated on, a capability only recently available on publicly accessible quantum computers. We implement the protocol in Ref [1] for tetrad POVM measurements—a minimal proof of concept—on IBM quantum computers using a single auxiliary qubit. We compare the binary tree approach to the Neumark extension method, and our preliminary results suggest that nontrivial POVMs can be measured with higher fidelity with the binary tree approach, demonstrating a promising application of mid-circuit measurements and feed-forward capabilities of near-term quantum devices.

1 Background

The two main cases of generalized measurements are the projective von Neumann measurement (PM) and the positive operator value (POVM) measurement. A projective measurement is described by Hermitian operators, $M = \sum_m mE_m$ on the state space of the system being measured. Here, E_m are projectors satisfying the conditions, $\sum E_m = I$, $E_mE_n = \delta_{mn}E_m$ and $E_m^\dagger = E_m$. The probability of outcome m occurring given that we measured the state $|\psi\rangle$ is given by Born's rule as $p(m) = \|E_m |\psi\rangle\|^2 = \langle\psi| E_m |\psi\rangle$. The post-measurement state is $\frac{E_m|\psi\rangle}{\|E_m|\psi\rangle\|}$.

POVMs, on the other hand, are in fact a generalization of projective measurements to larger space. They are equivalent to performing a projective measurement on the joint state of the system of interest coupled to an auxiliary system after a unitary operation. They can offer an advantage over projective measurements in quantum information processing tasks such as state descrimination. The set of measurement operators $\{E_a\}$ for a POVM are not projectors (i.e. $E_iE_j \neq \delta_{ij}E_i$), but are rather Hermitian ($E_a^\dagger = E_a$), positive ($\langle\psi| E_a |\psi\rangle \geq 0$), and complete ($\sum_a E_a = I$). The probability of measuring outcome a of the POVM is given as $p(a) = \langle\psi| E_a |\psi\rangle$.

2 Implementation and Experimental Results

Implementing POVM is non-trivial, and to this end, several methods have been proposed [1]. To compare contrasting approaches, we focus on two methods: Naimark's extension and the binary tree search methods, which importantly differ in terms of their scaling with the size of the POVM set and hardware capabilities necessary to implement them.

2.1 Naimark's extension

The Naimark's extension theory provides the path to implement generalized POVM measurement using projective von Neumann measurements by enlarging the Hilbert space of the measured system with an auxiliary quantum system as shown in Fig. 1. More formally, for every POVM $\{M_a\}_{a\in\mathcal{A}}$, there exists an isometry U such that

$$M_a = U^\dagger \left(\mathcal{I} \otimes |a\rangle \langle a|\right) U \; \forall \; a \in \mathcal{A}. \tag{1}$$

By implication, the isometry, U can be defined as

$$U = \sum_{a\in\mathcal{A}} M_a \otimes |a\rangle \tag{2}$$

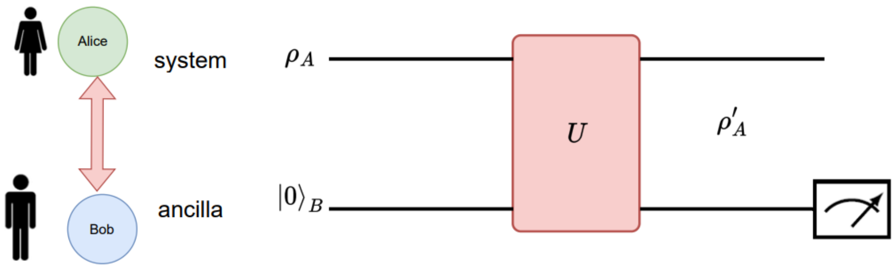

Figure 1: Example of Naimark's extension implementation. Auxilliary system $|0\rangle_B$ (also know as the ancilla) is introduced to the system of interest ρ_A. This is similar to bringing Bob to help Allice in measuring the POVM elements

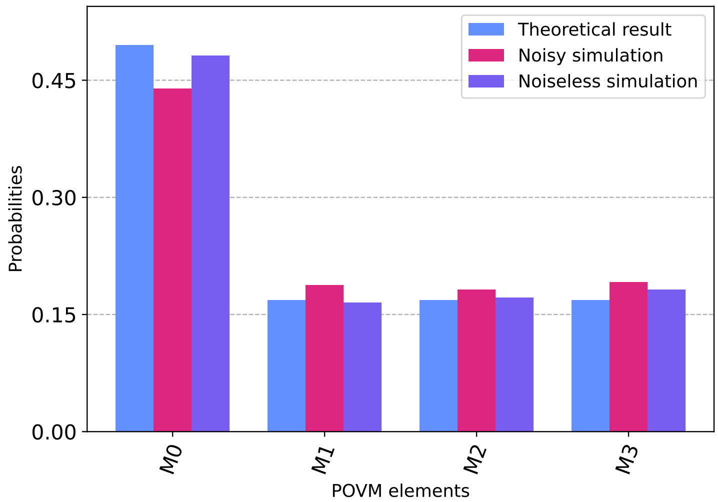

Figure 2: Probability distribution for the POVM elements. The labels on the X-axis show the different POVM elements. The noiseless simulation and theoretical result agree within shot noise, while the noisy simulation result qualitatively agrees with the ideal results.

We implemented the Naimark extension to determine the outcome of the tetrad POVM in [1] when the $|0\rangle$ state was measured and compared the result with the expected theoretical outcome of the probabilities of the POVM elements. Fig. 2 shows the probability distribution obtained for the Naimark extension and the theoretical results. We implemented the ideal noiseless case of the Naimark extension using IBM QasmSimulator and a noisy simulation of the Naimark extension using the ibm_perth noise model.

It may be challenging to implement Naimark's extension on current quantum computers with limited number of qubits because of its large auxiliary system requirement. Therefore, we consider a way to reduce the number of ancillary dimensions by making sequential measurements in the next section.

2.2 Binary tree approach

We implemented the binary tree algorithm for a POVM of an arbitrary dimension. As a reminder, the binary tree algorithm described in Ref.[1] requires a single ancilla qubit for its implementation. It performs a binary search by iteratively applying unitaries conditioned on the results of mid-circuit measurements. The implementation in the form of a Python module consists of the central class BinaryTreePOVM and several helper functions. It creates a simple way to construct a measurement circuit for an arbitrary POVM. For example, Fig. 3 shows the corresponding measurement circuit for the tetrad POVM on a single qubit system. With the BinaryTreePOVM class, one can create this circuit in two lines of code:

```python
from binary_tree_povm import *
# given a valid POVM M
btp = BinaryTreePOVM(M)
btp.qc.draw('mpl')
```

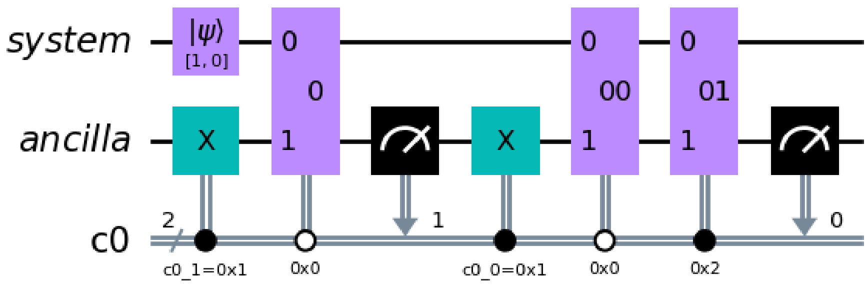

Figure 3: Binary tree measurement circuit for the tetrad POVM acting on a single qubit system

We tested the implementation on several randomly generated POVMs of different dimensions. The number of POVM elements was also varied. In the following experiment, we used the tetrad POVM derived in Ref.[1]. Tetrad POVM is a four-element POVM acting on a system consisting of a single qubit. Fig. 4 shows probability distributions for different simulations along with theoretically expected results.

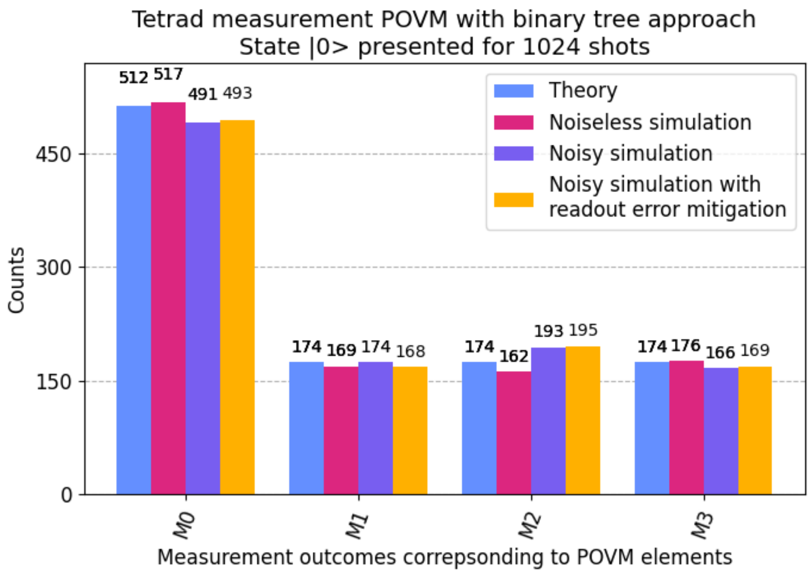

Figure 4: Probability distribution for measuring different POVM elements on noisy and noiseless simulators. POVM elements are denoted as M0, M1, M2 and M3.

In Fig. 4, we can see that the simulated probability distribution strongly correlates with the theoretical values. Additionally, we investigate the effect of noise by transpiling the circuit to the topology of ibm_perth, the only backend available to our team that supports dynamic circuits. We initialized the QasmSimulator with ibm_perth as a backend for the noisy simulator. This way, the actual hardware's noise model can be mimicked. However, dynamic operations, i.e., gates conditioned upon the state of the classical register, are not mimicked by the noise model. Therefore, one can only assess the impact of noise in these operations based on hardware experiments. However, we could assess the noise from other operations by comparing noisy and noiseless simulations. We observe that the noisy and noiseless simulation results closely agree, which indicates moderate noise. Further, we could evaluate the impact of measurement errors by applying readout error mitigation to the noisy results. Fig. 4 shows that readout errors do not distort the probability distribution significantly because error-mitigated results are almost identical to the noisy results.

3 Outlook

Our next step will be to obtain hardware results from ibm_perth for both Naimark and binary tree implementations. Further, we need to develop a metric(s) based on which these results will be assessed. For example, one can analyze the closeness of hardware results to the theoretical expectations. It would also be interesting to compare the circuit depths of the transpiled circuits of the two algorithms. For example, we will investigate the dependence of circuit depth on the number of POVM elements and the size of the system to understand how device noise affects the quality of the generalized measurements. We also plan to optimize the transpilation process, especially the decomposition into native gates, to further decrease the circuit depth for both algorithms.

References

[1] Erika Andersson and Daniel KL Oi. Binary search trees for generalized measurements. *Physical Review A*, 77(5):052104, 2008.