Non-Boolean Quantum Amplitude Amplification and ML applications

Prasanth Shyamsundar, Fermilab Quantum Institute

2021 Snowmass Workshop on Quantum Computing for HEP

December 03, 2021

Machine Learning

A high-level view of Machine Learning:

1. Design a highly parameterized machine to perform a task.
   (architecture, trainable weights)
2. Design a way to evaluate the performance of the machine (cost function).
   Often evaluated by averaging a loss function over the training data.
3. Train the machine by optimizing the cost function.

In typical hybrid-quantum-classical ML approaches

- (1) is done with **quantum** circuits parameterized by **classical** parameters
- (2) and (3) are performed **classically**

Can we do better in the post-NISQ era?

In this talk, I will lay the foundations for an inherently quantum technique for ML training.

Introduction: Boolean Amplitude Amplification

Given:

- A quantum system with basis states $|0\rangle, |1\rangle, \dots, |N-1\rangle$
- A Boolean function $f: \{0, \dots, N-1\} \rightarrow \{0,1\}$ ("bad"/"good")
- An initial superposition of the states $|\psi_0\rangle = A_0|0\rangle$
- An oracle for the function $U_f|x\rangle = (-1)^{f(x)}|x\rangle$

Goal: Amplify the "good" states

$$|\psi_0\rangle = \cos(\theta/2)\,|\psi_{bad}\rangle + \sin(\theta/2)\,|\psi_{good}\rangle$$

$|\psi_{bad}\rangle$ and $|\psi_{good}\rangle$ are the good and bad projections of $|\psi_0\rangle$, normalized to 1.

Amplitude Amplification: Apply "$S\ U_f$" repeatedly on $|\psi_0\rangle$. After k steps

$$|\psi_k\rangle = \cos(k\theta + \theta/2)\,|\psi_{bad}\rangle + \sin(k\theta + \theta/2)\,|\psi_{good}\rangle$$

We could use this to optimize/train a machine.

But the performance measure of machines is unlikely to be Boolean!

Non-Boolean Amplitude Amplification

Let's make the function Non-Boolean:

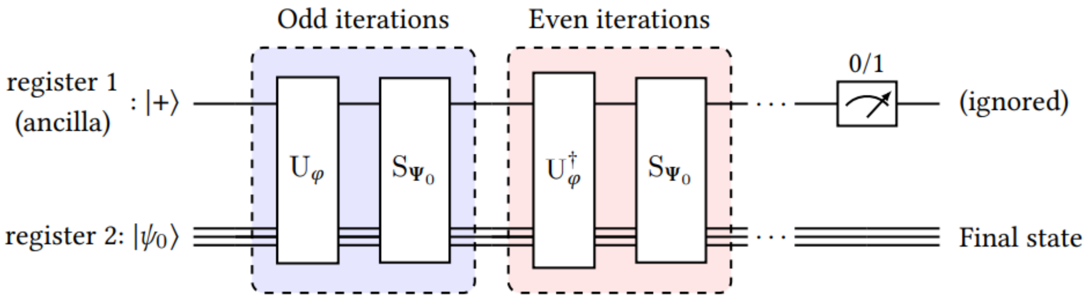
$$\varphi: \{0, \ldots, N-1\} \to \mathbb{R}$$
$$U_\varphi |x\rangle = e^{i\varphi(x)} |x\rangle$$

"Goodness" of a state is captured by $\cos\varphi(x)$.

Goal: Preferentially amplify states, based on their goodness.

The original amplitude algorithm will not work in this case.

The non-Boolean algorithm introduced in
P. Shyamsundar, [arXiv:2102.04975](https://arxiv.org/abs/2102.04975) [quant-ph] will work!



Fermilab

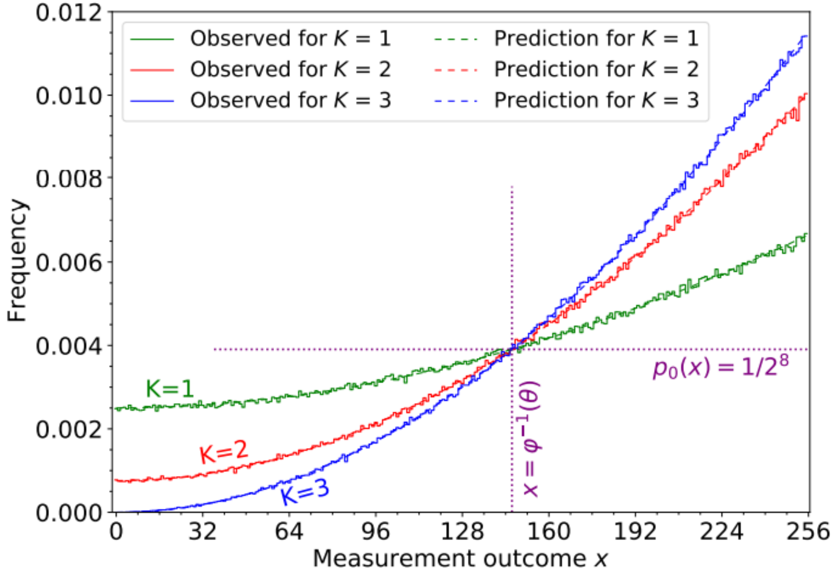
Non-Boolean Amplitude Amplification in Action

A toy example

$$\varphi(x) = \frac{x}{255}\frac{\pi}{4}, \qquad \text{for } x = 0,1,\ldots,255$$

$$U_\varphi\left|x\right\rangle = e^{i\varphi(x)}\left|x\right\rangle$$

Amplified state measurement probabilities after 1, 2, and 3 iterations:



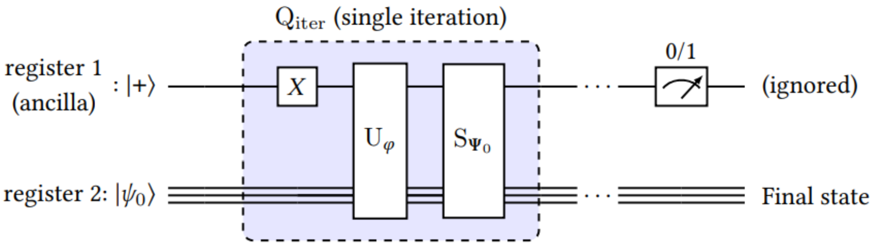
Fermilab

Quantum Mean Estimation Algorithm

Goal: Estimate $\langle\psi_0|U_\varphi|\psi_0\rangle$. Let $\cos\theta = Re\,\langle\psi_0|U_\varphi|\psi_0\rangle$

$$|+,\psi_0\rangle = \frac{|\eta_+\rangle + |\eta_-\rangle}{\sqrt{2}}$$

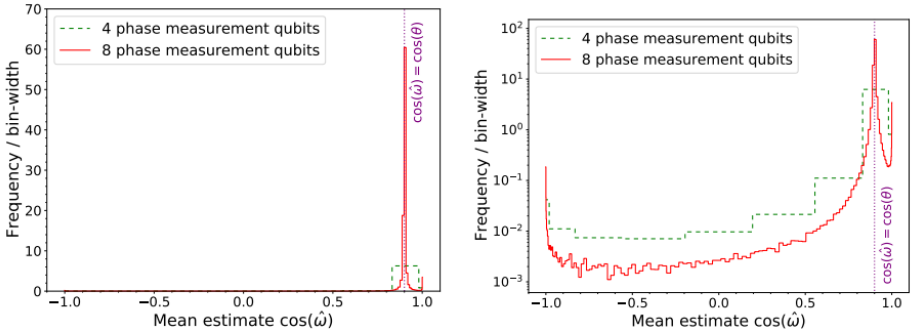
$$Q_{\text{iter}}|+,\psi_0\rangle = \frac{e^{i\theta}|\eta_+\rangle + e^{-i\theta}|\eta_-\rangle}{\sqrt{2}}$$



Circit from [arXiv:2102.04975](https://arxiv.org/abs/2102.04975) [quant-ph]

Quantum phase estimation can be used with Q_{iter} as the operator, and $|+,\psi_0\rangle$ as the state to estimate θ.

Toy example results:



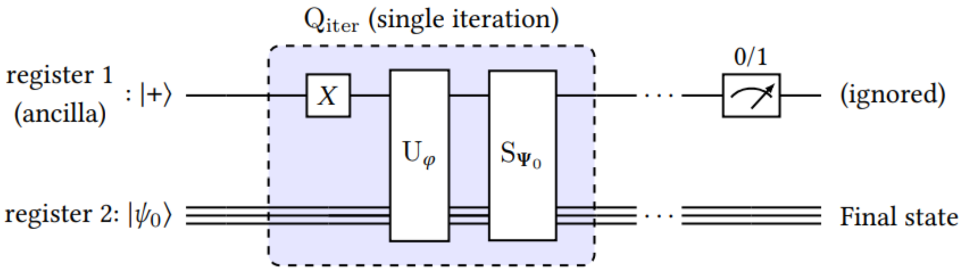
Fermilab

Quantum Mean Estimation Algorithm

Goal: Estimate $\langle\psi_0|U_\varphi|\psi_0\rangle$. Let $\cos\theta = Re\,\langle\psi_0|U_\varphi|\psi_0\rangle$

$$|+,\psi_0\rangle = \frac{|\eta_+\rangle + |\eta_-\rangle}{\sqrt{2}}$$

$$Q_{\text{iter}}|+,\psi_0\rangle = \frac{e^{i\theta}|\eta_+\rangle + e^{-i\theta}|\eta_-\rangle}{\sqrt{2}}$$



1. Quadratic speed-up over shot-based estimation.
2. Can be used in ML, for example to average over training data.

Fermilab

Back to Machine Learning

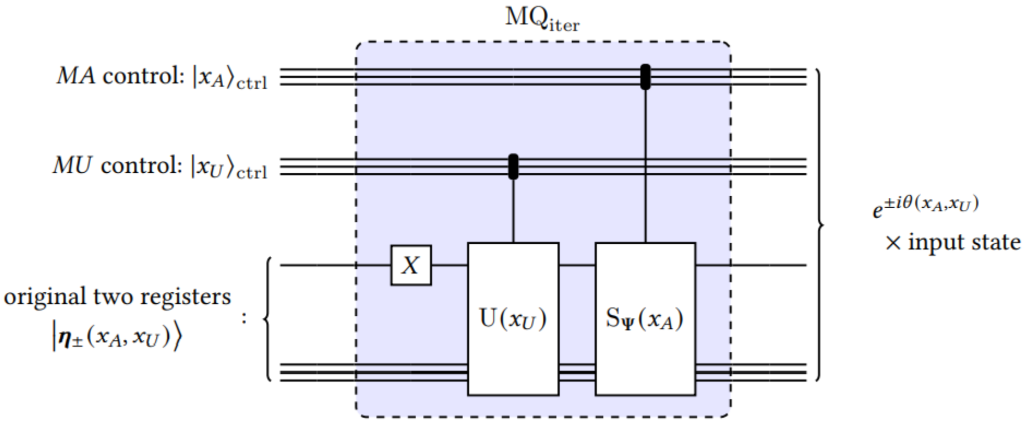
Training a machine:

Optimize[Avg[Loss for one data sample]]

Non-Boolean amplification

Operator use in Quantum Mean Estimation

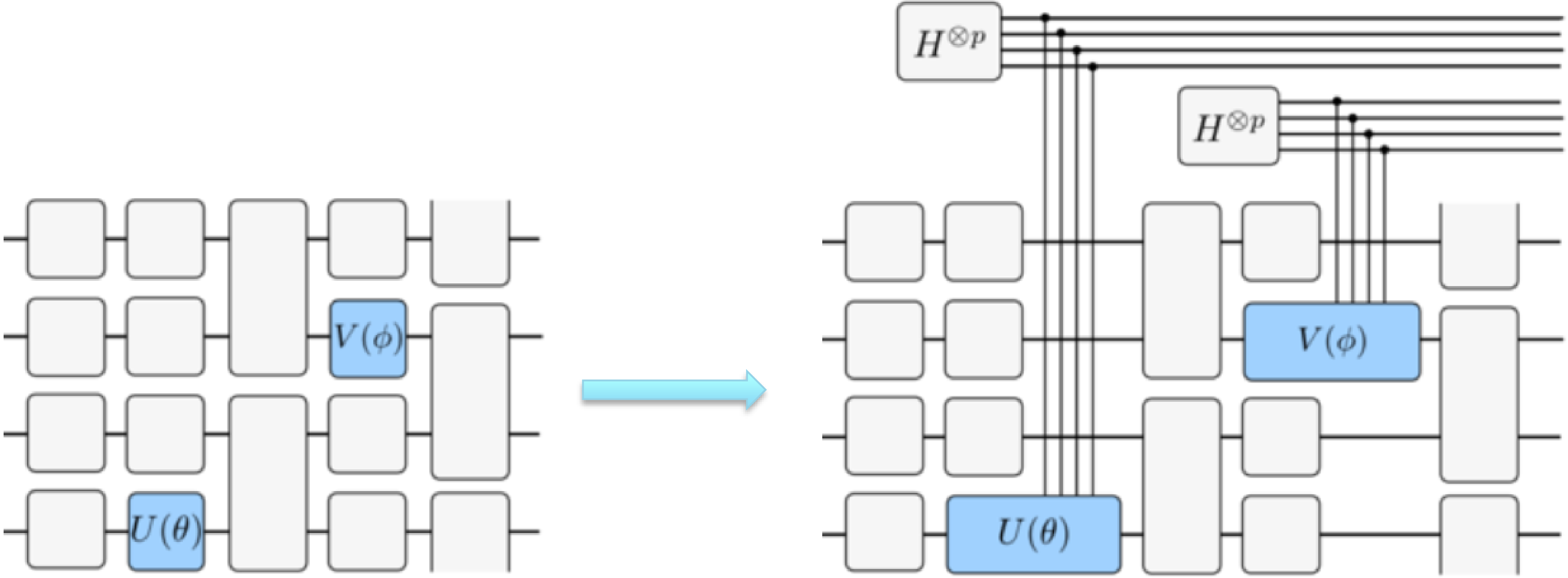
Accomplished in this circuit:



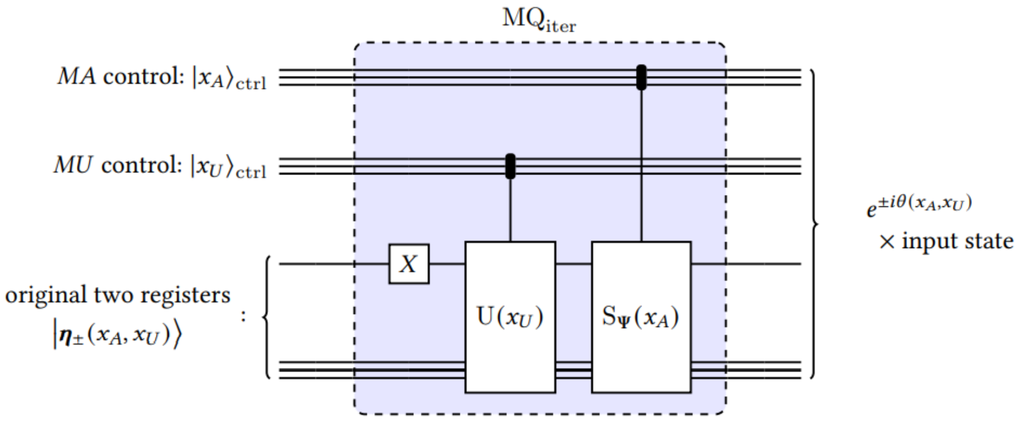
Fermilab

Final piece of the puzzle

Qarameterized circuits: Quantum circuits parameterized by **qubits**



Example QML approaches



Example 1: Classifier parameterized by $|x_U\rangle$

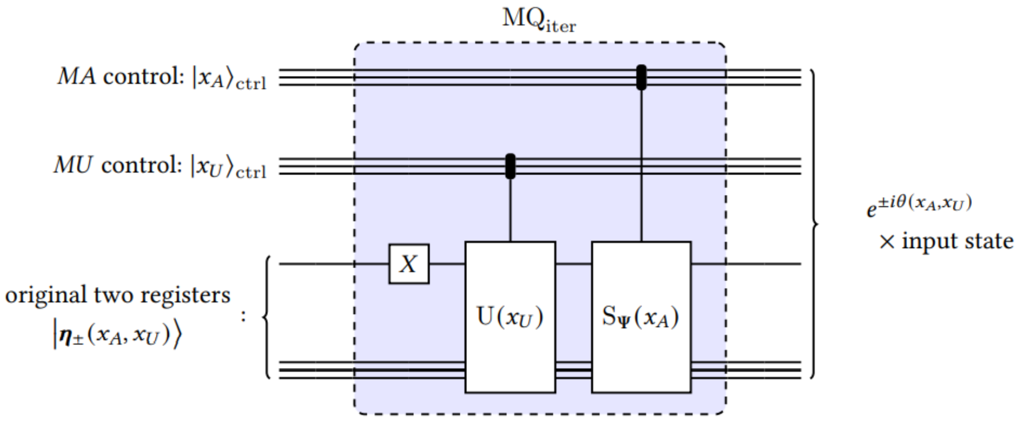
4th register: superposition of all training datasamples

$U(x_U)$: output of classifier CNOT-ed with the target

θ captures the average classification accuracy of the classifier.

Training can be done using amplitude amplification!

Example QML approaches



Example 2: State preparation circuit parameterized by $|x_A\rangle$

4th register: $|0\rangle$

U: Evaluates the prepared state.

This circuit produces an evaluation oracle for $|x_A\rangle$.
Training can be done using amplitude amplification!

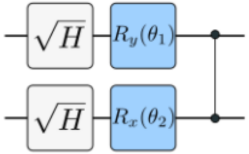
Results from QHACK 2021

Work done in collaboration with **Evan Peters, University of Waterloo & FQI**

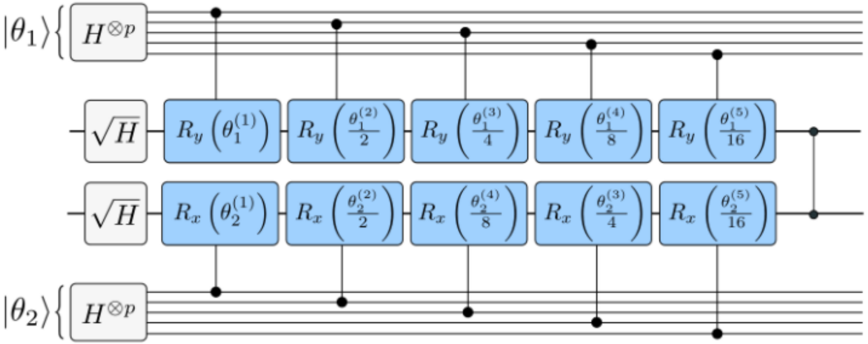
Presentation at https://peterse.github.io/groveropt/

Parameterizing circuits with qubits aka Qarameterized circuits:

Classical Parameterized circuit



Lifted Qarameterized circuit

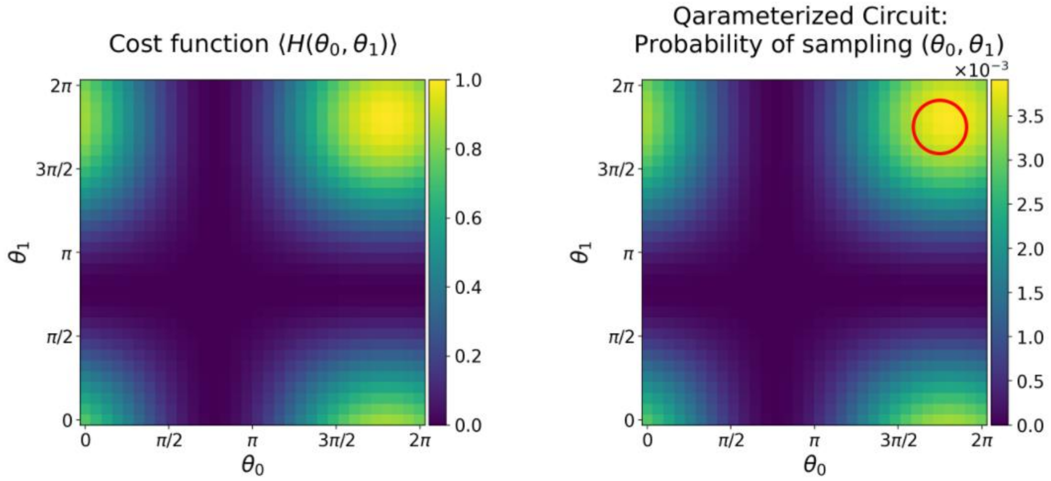


Fermilab

Results from QHACK 2021

Left panel: Cost function

Right panel: Sampling probability of the parameters, post training using non-Boolean amplification



Thank you!

Acknowledgements

U.S. DEPARTMENT OF ENERGY | Office of Science

This research is supported by the U.S. Department of Energy, Office of Science, Office of High Energy Physics QuantISED program under the grant a) “HEP Machine Learning and Optimization Go Quantum”, Award Number 0000240323, and b) “DOE QuantiSED Consortium QCCFP-QMLQCF”, Award Number DE-SC0019219.