Variational Quantum Classification with QiSkit & Developments in Particle Tracking with D:Wave

Eric Drechsler, Parker Reid
Simon Fraser University

30. 10. 2019
Quantum Computing Mini-Workshop
LBNL/Berkeley

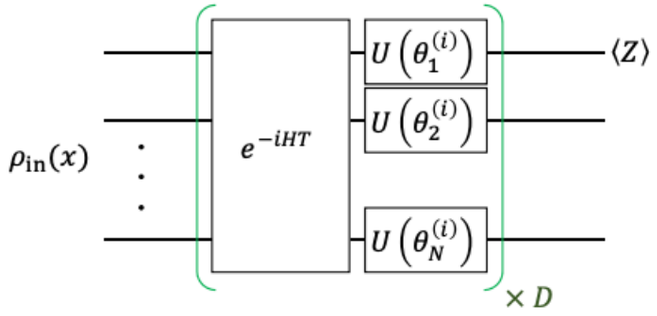
Special Thanks to: Olivia di Matteo (TRIUMF)

SFU SIMON FRASER UNIVERSITY
ENGAGING THE WORLD

Variational Quantum Classification (VQC) with QiSkit

Qulacs

Project page



Mitarai, K. et al.: *Quantum Circuit Learning*

arxiv:1803.00745

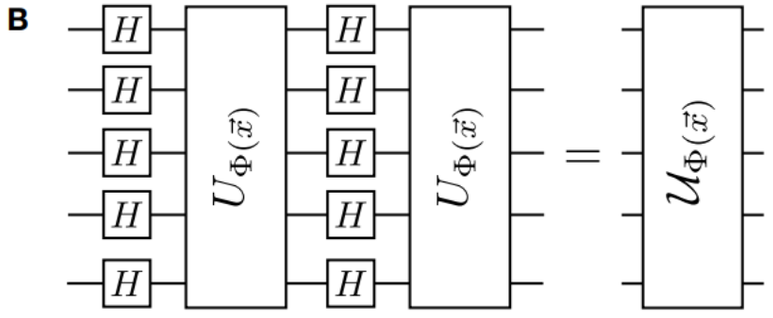
See Ryu Sawada's Presentation

Qiskit

An open-source framework for working with noisy quantum computers

https://qiskit.org qiskit@qiskit.org

VQC implementation



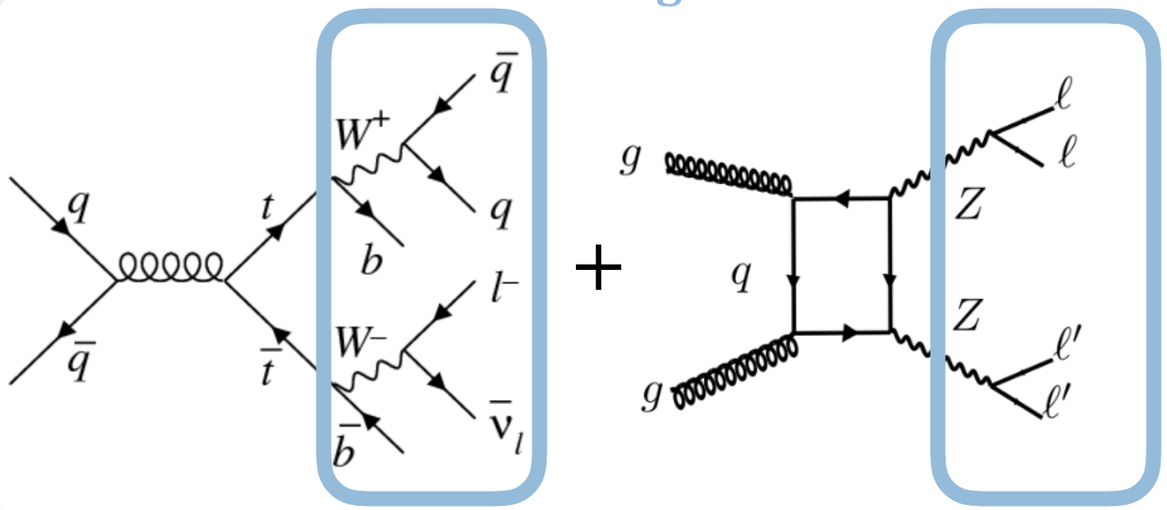
Havlicek, V. et al.: *Supervised learning with quantum enhanced feature spaces*
Nature 567, 209–212 (2019)
arxiv:1804.11326

Special Thanks to: Koji Terashi (ICEPP, Tokyo)

Example HEP Classification Problem

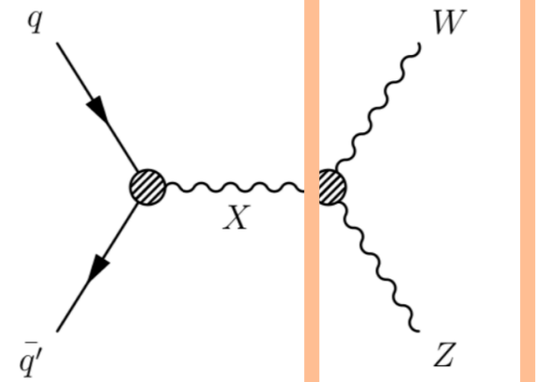
Eric Drechsler, 30. Oct. 2019

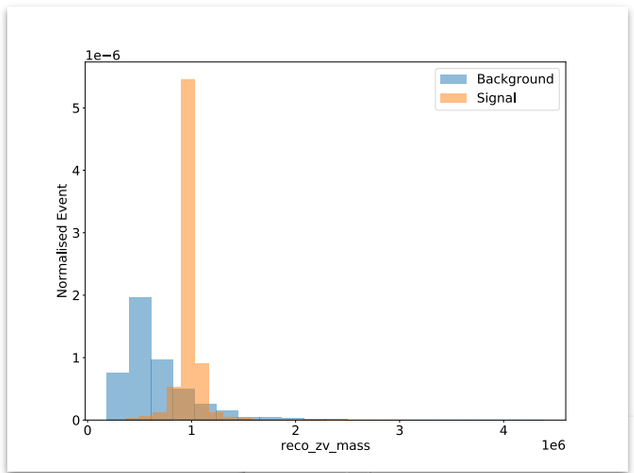
Background

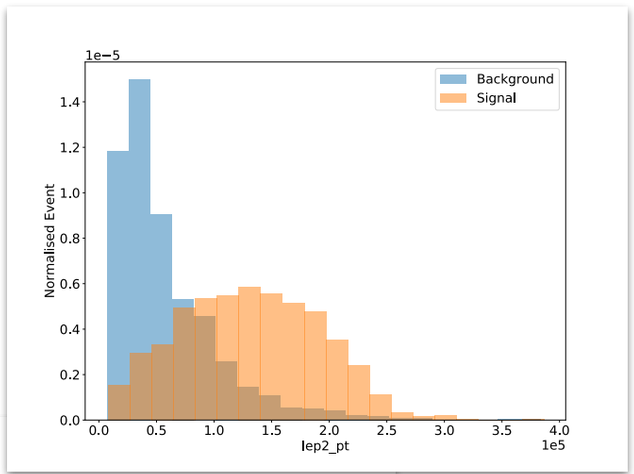


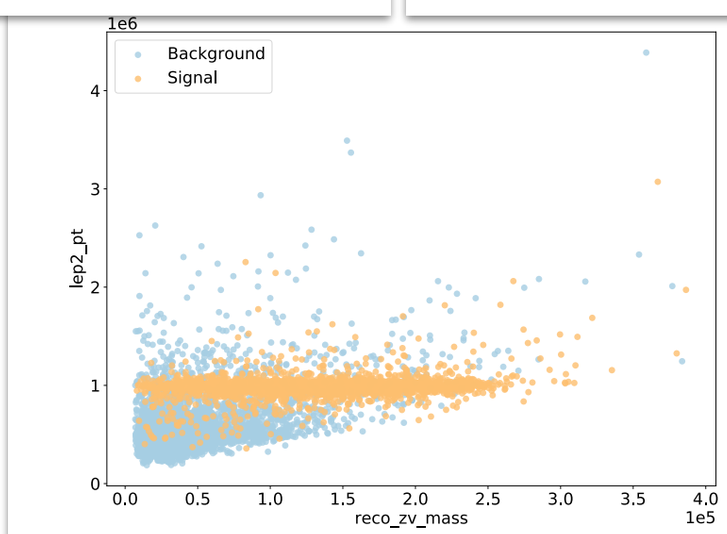
vs.

Signal









Source Dataset

Train

Unlabelled Data

Classifier

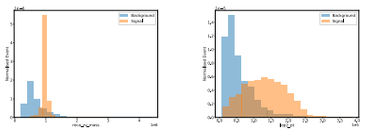
Background

Signal

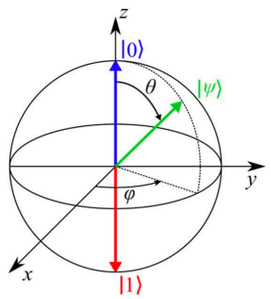
VQC: Algorithm Overview

Eric Drechsler, 30. Oct. 2019

Input Data



Step I.
Create Quantum State



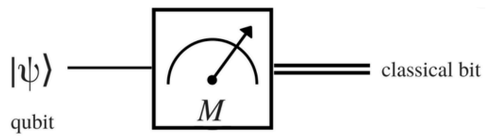
Feature Map

Step II.
Parameterise State

$$|\psi\rangle \quad U3(\theta,\phi,\lambda) \quad U(\theta,\phi,\lambda)\,|\psi\rangle$$

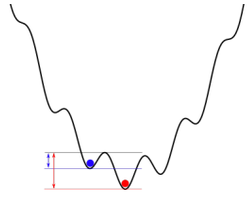
Variational Form

Step III.
Measurement



Expectation Value

Step IV.
Classical minimization algorithm



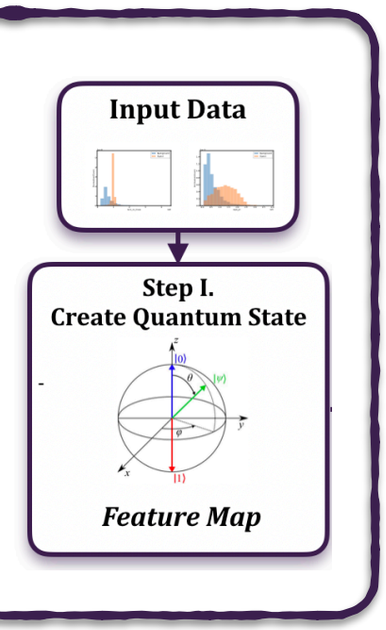
Update Parameters

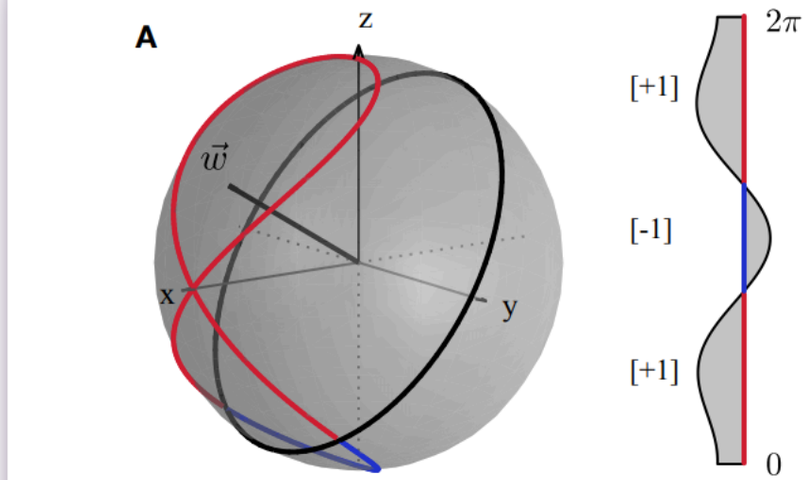
$(\theta_{i+1}, \phi_{i+1}, \lambda_{i+1})$

$< \psi_i|M|\psi_i >$

VQC: Input Data to Quantum State

Eric Drechsler, 30. Oct. 2019



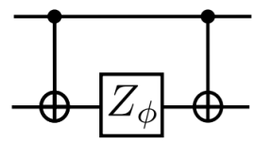


Feature map generator:

$$\mathcal{U}_\Phi(\vec{x}) = U_{\Phi(\vec{x})} H^{\otimes n} U_{\Phi(\vec{x})} H^{\otimes n}$$

$$U_{\Phi(\vec{x})} = \exp\left(i \sum_{S \subseteq [n]} \phi_S(\vec{x}) \prod_{i \in S} Z_i \right)$$

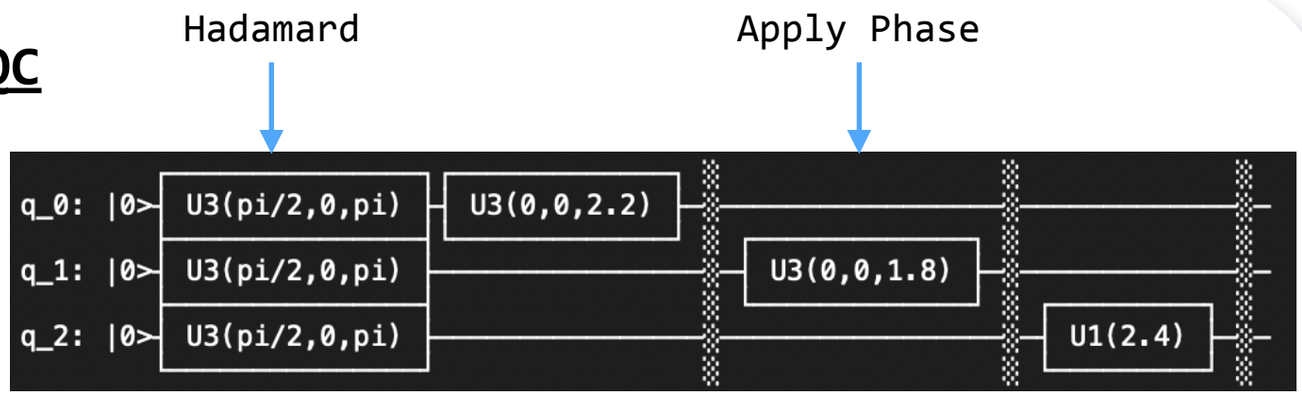
$$e^{i\phi_{\{l,m\}}(\vec{x}) Z_l Z_m} =$$



- **QiSkit implementations:**
- **FirstOrderExpansion**
- SecondOrderExpansion
- amongst <u>others</u>

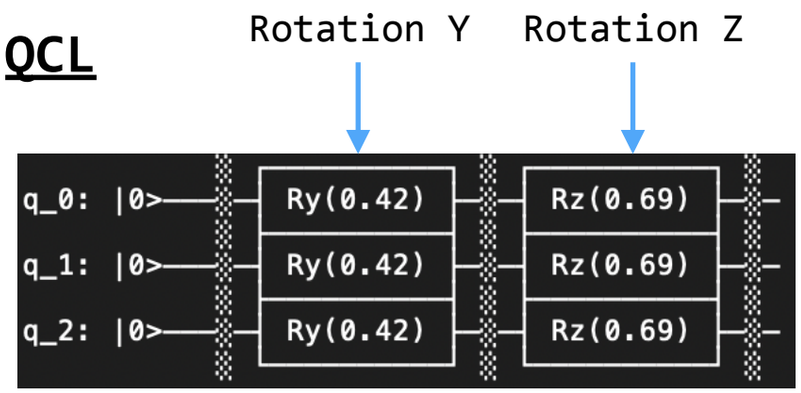
$$S \in \{0, 1, ..., n-1\}, \phi_i(\vec{x}) = x_i$$

VQC



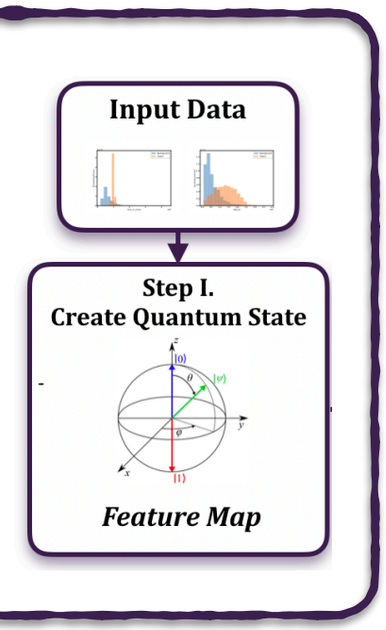
$$q_0 : |0\rangle \rightarrow \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \rightarrow \frac{1}{\sqrt{2}}(e^{ix}|0\rangle + |1\rangle)$$

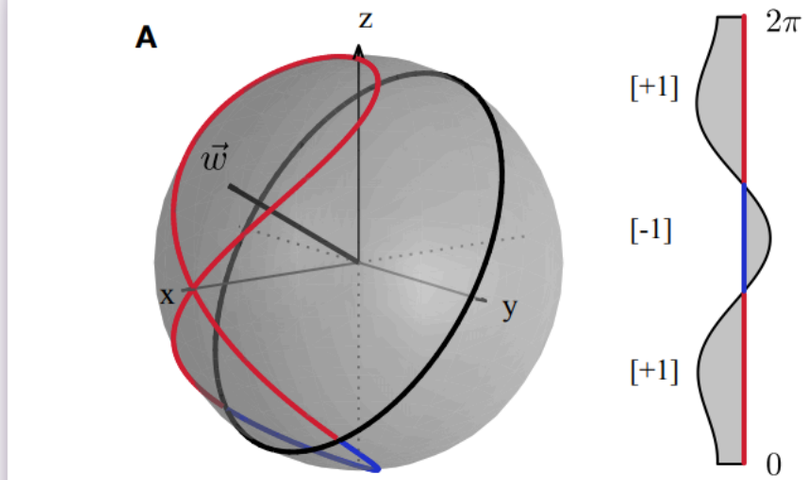
QCL



VQC: Input Data to Quantum State

Eric Drechsler, 30. Oct. 2019



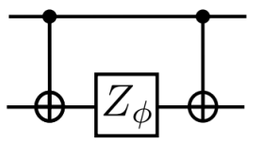


Feature map generator:

$$\mathcal{U}_\Phi(\vec{x}) = U_{\Phi(\vec{x})} H^{\otimes n} U_{\Phi(\vec{x})} H^{\otimes n}$$

$$U_{\Phi(\vec{x})} = \exp\left(i \sum_{S \subseteq [n]} \phi_S(\vec{x}) \prod_{i \in S} Z_i \right)$$

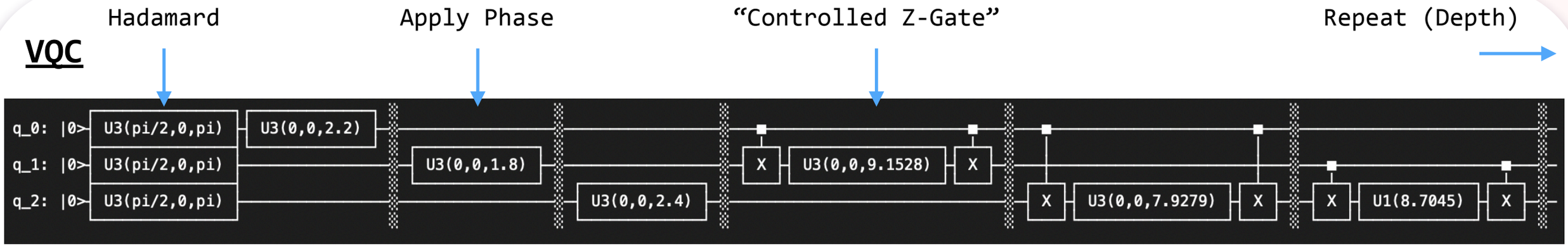
$$e^{i\phi_{\{l,m\}}(\vec{x}) Z_l Z_m} =$$



- QiSkit implementations:
  - FirstOrderExpansion
  - **SecondOrderExpansion**
  - amongst others

$$S \in \{0, 1, ..., n-1, (0,1), (0,2), ..., (n-2, n-1)\},$$
$$\phi_i(\vec{x}) = x_i, \phi_{(i,j)}(\vec{x}) = (\pi - x_i) * (\pi - x_j)$$

VQC



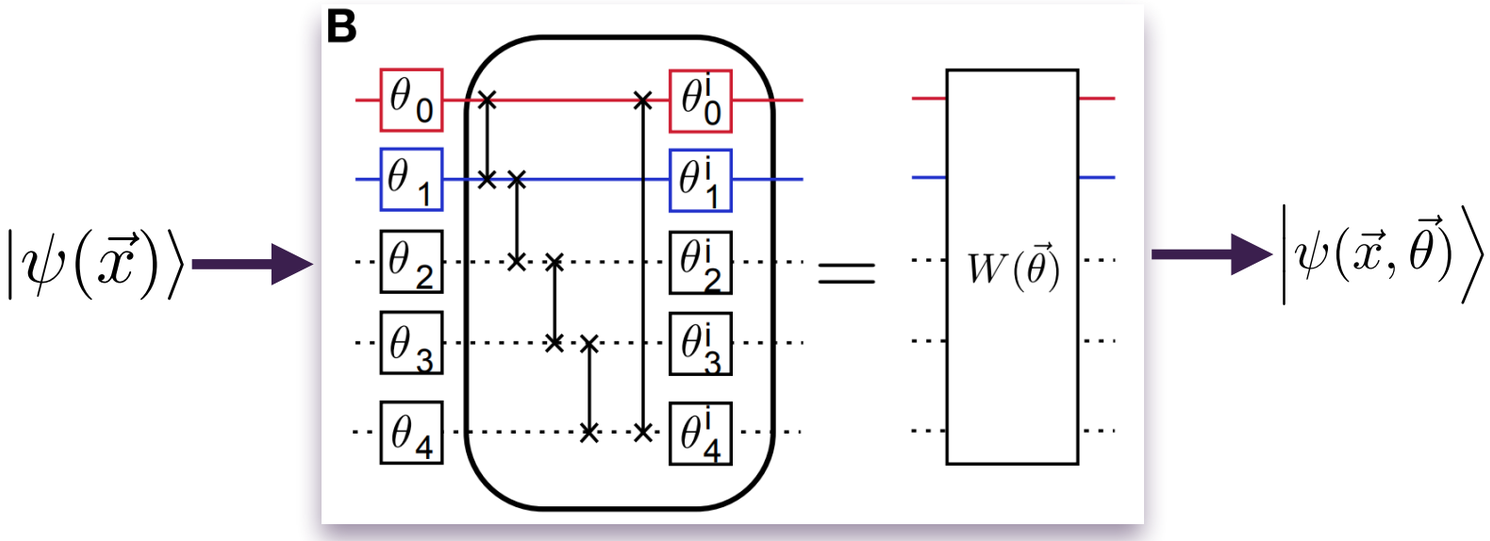
$$q_0 : |0\rangle \rightarrow \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \rightarrow \frac{1}{\sqrt{2}}(e^{ix}|0\rangle + |1\rangle)$$
$$q_1 : |0\rangle \rightarrow \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \rightarrow \frac{1}{\sqrt{2}}(e^{iy}|0\rangle + |1\rangle)$$

$$|q_1 q_2\rangle \rightarrow \frac{1}{2}(e^{i(x+y+\alpha)}|00\rangle + e^{ix}|01\rangle + e^{iy}|10\rangle + e^{i\alpha}|11\rangle)$$

Step II.
Parameterise State

$|\psi\rangle$ — $U3(\theta, \phi, \lambda)$ — $U(\theta, \phi, \lambda)\,|\psi\rangle$

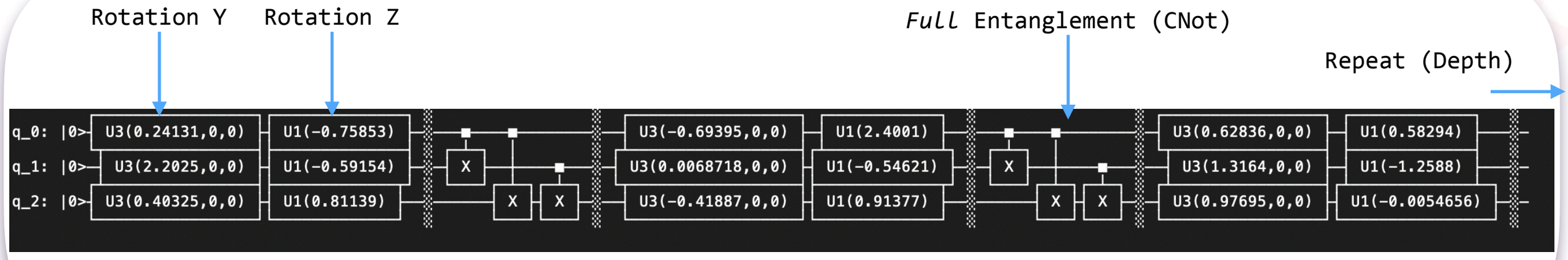
Variational Form



- **QiSkit Variational Forms:**
  - Ry - Rotations Y + Entanglement
  - RyRz -Rotations Y, Z + Entanglement
  - SwapRz, UCSSD

- Number of Parameters for **RyRz** VF:
  - Number of Qbits q
  - Depth d

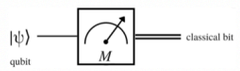
$$q \times (d+1) \times 2$$



VQC: Expectation Value Measurement

Eric Drechsler, 30. Oct. 2019

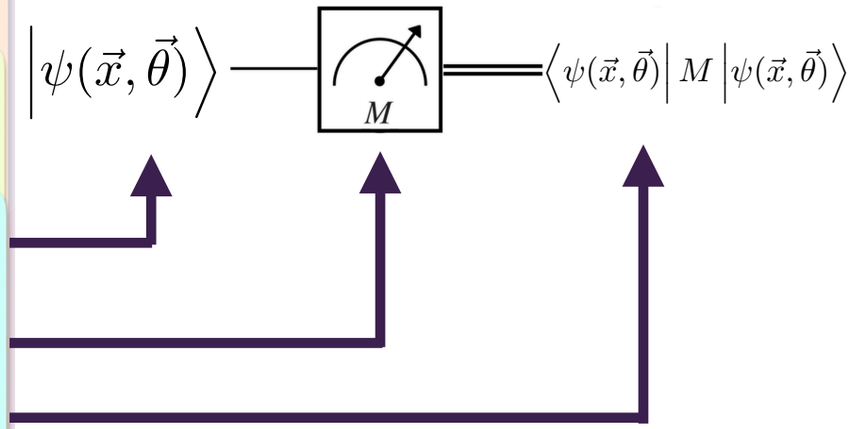
Step III.
Measurement

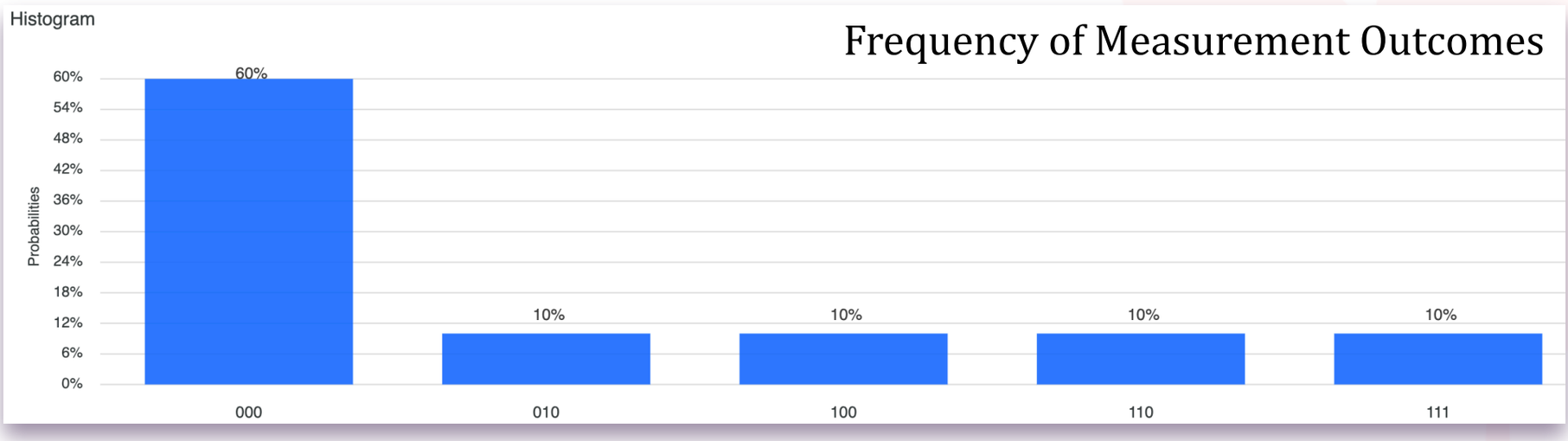


Expectation Value

```
for x in Dataset:
    measurements=[]
    for shot in #shots:
        prepareState
        applyVariationalForm
        m=measureCircuit
        measurements[shot]=m
    expectationValue=average(measurement)
```

QPU

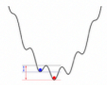




VQC: Input Data to Quantum State

Eric Drechsler, 30. Oct. 2019

Step IV.
Classical minimization algorithm

$(\theta_{i+1}, \phi_{i+1}, \lambda_{i+1})$  $< \psi_i|M|\psi_i >$

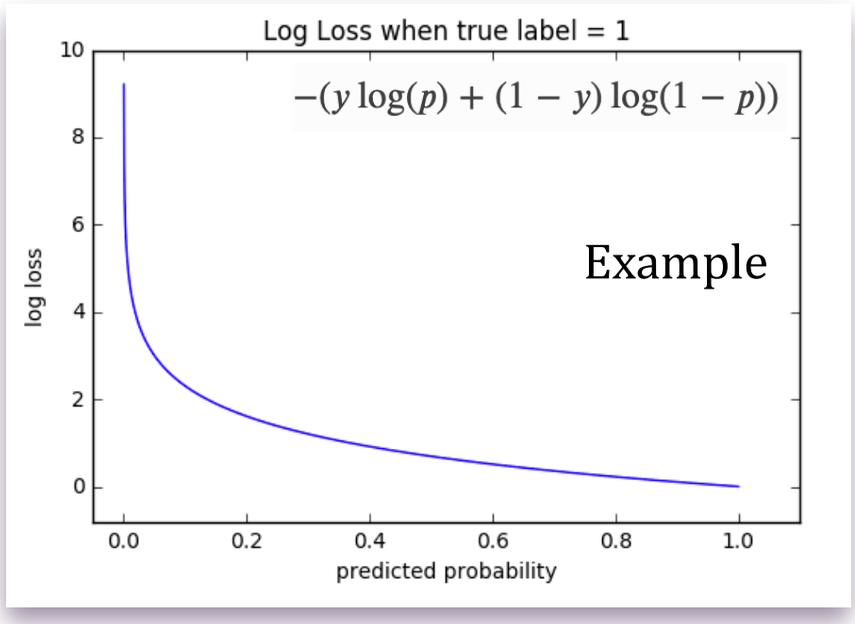
Update Parameters

```
for step in optimisationsStep:
    for x in Dataset:
        measurements=[]
        for shot in #shots:
            prepareState
            applyVariationalForm
            m=measureCircuit
            measurements[shot]=m          QPU
        expectationValue=average(measurement)
    updateParameters(expValue,step)
```

Qiskit Default Optimisers

ADAM ([maxiter, tol, lr, beta_1, beta_2, ...])	Adam Kingma, Diederik & Ba, Jimmy.	
CG ([maxiter, disp, gtol, tol, eps])	Conjugate Gradient algorithm.	
COBYLA ([maxiter, disp, rhobeg, tol])	Constrained Optimization By Linear Approximation algorithm.	
L_BFGS_B ([maxfun, maxiter, factr, iprint, ...])	Limited-memory BFGS algorithm.	tested
NELDER_MEAD ([maxiter, maxfev, disp, xatol, ...])	Nelder-Mead algorithm.	
P_BFGS ([maxfun, factr, iprint, max_processes])	Limited-memory BFGS algorithm.	
POWELL ([maxiter, maxfev, disp, xtol, tol])	Powell algorithm.	
SLSQP ([maxiter, disp, ftol, tol, eps])	Sequential Least SQuares Programming algorithm	
SPSA ([max_trials, save_steps, last_avg, c0, ...])	Simultaneous Perturbation Stochastic Approximation algorithm.	recommended by authors
TNC ([maxiter, disp, accuracy, ftol, xtol, ...])	Truncated Newton (TNC) algorithm.	
AQGD ([maxiter, eta, tol, disp, momentum])	Analytic Quantum Gradient Descent (AQGD) optimizer class.	

Qiskit Loss Function



Loss function used by authors:

Empirical Risk:

$$R_{\text{emp}}(\vec{\theta}) = \frac{1}{|T|} \sum_{\vec{x} \in T} \Pr\left(\tilde{m}(\vec{x}) \neq m(\vec{x})\right)$$

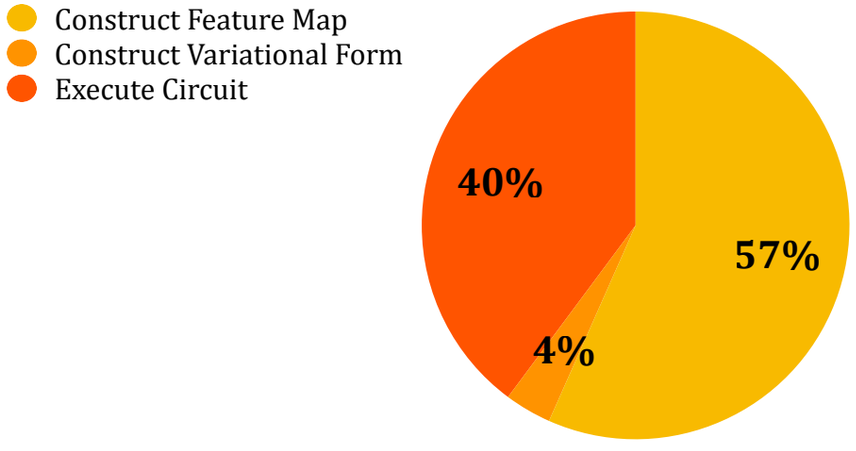
$$\Pr\left(\tilde{m}(\vec{x}) \neq m(\vec{x})\right) \approx \text{sig}\left(\frac{\sqrt{R}\left(\frac{1}{2} - \left(\hat{p}_y(\vec{x}) - \frac{yb}{2}\right)\right)}{\sqrt{2(1 - \hat{p}_y(\vec{x}))\hat{p}_y(\vec{x})}}\right)$$

VQC: Results of Testruns

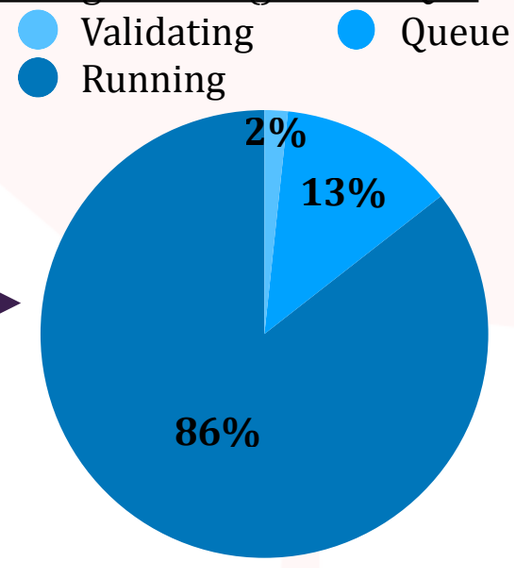
Eric Drechsler, 30. Oct. 2019

		VQC (Sim)	VQC (Sim)	VQC (IBMQX2)	QCL	DNN
	Comment	SecondOrderExp.	FirstOrderExp.	Testrun		Keras
Parameters	**Depth Feature Map**	2	2	2	-	
	Depth Variational Form	3	3	3	3	
	Optimiser	COBYLA	COBYLA	COBYLA	BFGS	
	#Events	1000	1000	10	1000	
	#Optimisations	50	50	1	100	
	#Shots	4096	4096	4096	-	
Timings	**Feature Map**	4458.8	2840.7	2.0		
[sec]	**Variational Form**	278.6	192.0	0.1		
	Execution	3135.7	1051.1	227.7		
	Training [hh:mm:ss]	02:12:16	01:07:46		00:45:41	00:00:29
Results	**Signal Efficiency**	**0.91**	0.86	-	0.93	0.91
	Testing Accuracy	0.83	0.79	-	-	-
	False Positive Rate	**0.17**	0.21	-	0.23	-

Timing Train() in VQC Simulation

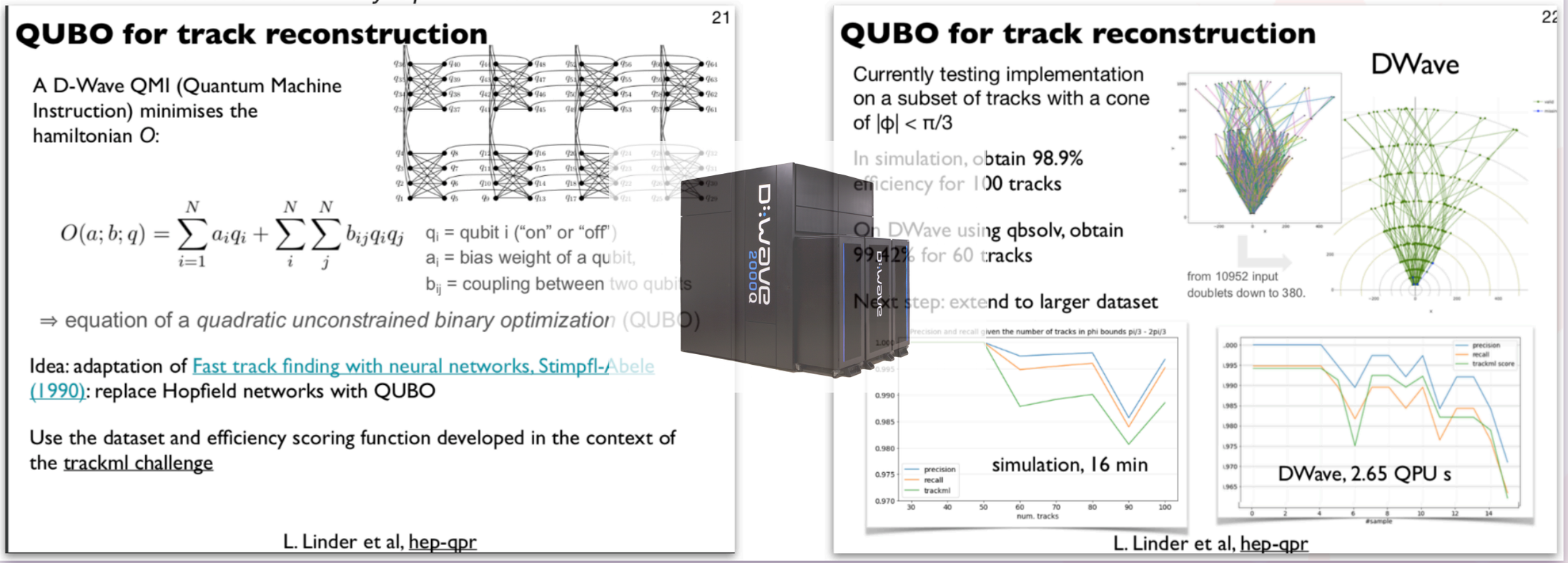


Timing Execute() on IBMQX2



Developments in Particle Tracking with D:Wave

Quadratic Unconstrained Binary Optimization



Work based on: Bapst, F. et al.: *A pattern recognition algorithm for quantum annealers* arxiv:1902.08324

hepqpr-project

qallse-package

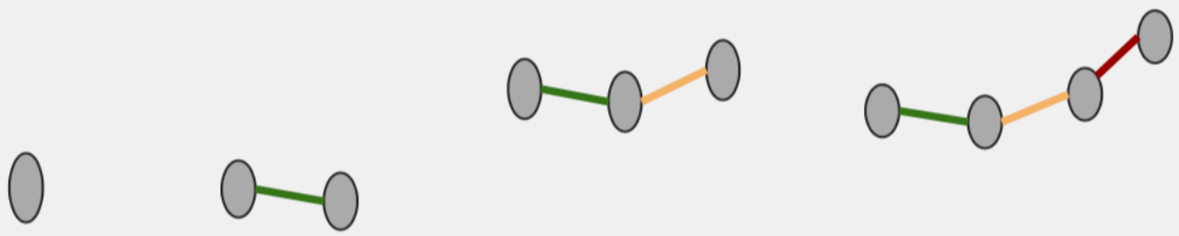
Special Thanks to:
Heather Gray, Lucy Lindner (LBNL, Berkeley)

Particle Tracking on a Quantum Annealer:

Quantum annealer allows for a NP-Hard minimization problem to be solved in a single adiabatic transition.

Particle tracks can be formed into **Q**uadratic **U**nconstrained **B**inary **O**ptimization problem.

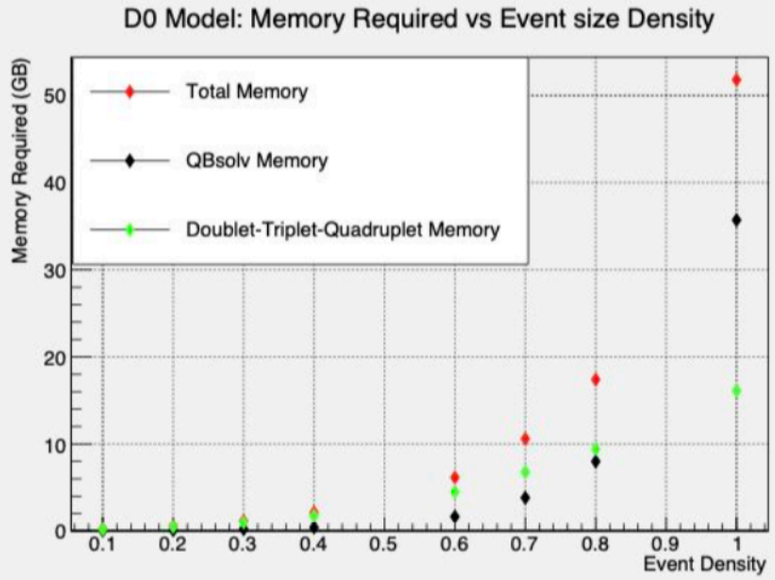
Hits in detectors -> doublets -> triplets -> quadruplets -> **QUBO** -> Tracks

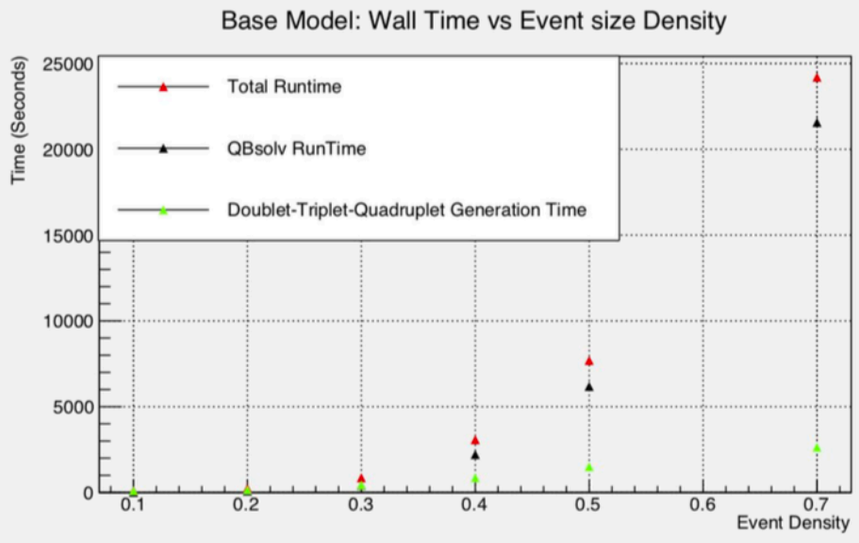


Based on previous work and software package from Lucy Linder (hepqrqallse: github.com/derlin/hepqpr-qallse)

Current QUBO's much too large to put on quantum computer
-> apply quantum classical hybrid algorithm (TABU search)

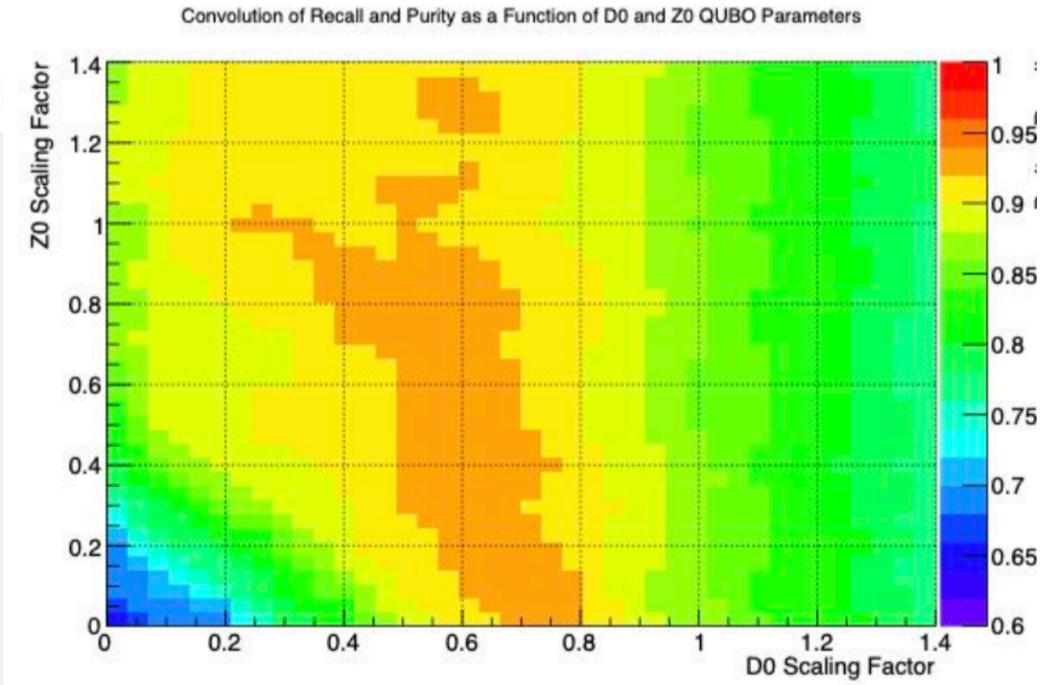
Memory and runtime deficiencies associated with hybrid algorithm





Parker Reid, 2019

$$a_i = \alpha(1 - e^{\frac{|d_0|}{\gamma}}) + \beta(1 - e^{\frac{|z_0|}{\lambda}})$$

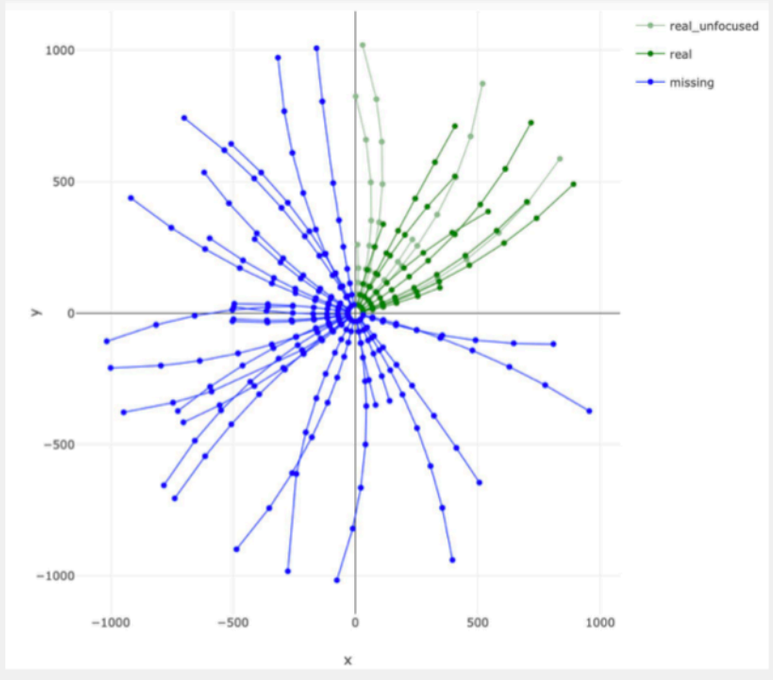


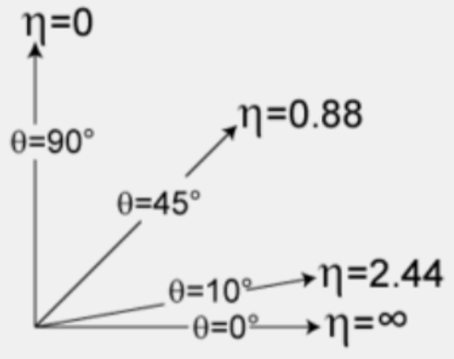
Ongoing Studies in Master's Project:

- **Parameter scans** on modified QUBO
  - improvement by using impact parameters?
- **Slicing** data w.r.t transverse angle **Φ** (x,y plane) pseudorapidity **η** (r,z plane)**:**
  - Data slices with overlapping regions

(constrained by maximum triplet curvature)

- Redundant tracks post QUBO require sorting
- Systematic error of overlap requires study
- Optimistic about algorithm speedup!





Parker Reid, 2019

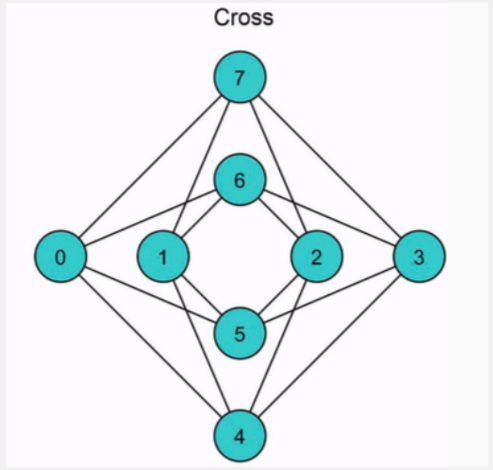
Planned Studies in Master's Project:

- QUBO modifications: setting biases in QUBO
- efficiency on dataset with secondary vertex contributions
  - Current dataset vertex at origin
  - using ACTS framework for data generation

Chimera:

6 couplings/qbit

2048 qbits (DW2000Q)



Planned: Study of Architecture

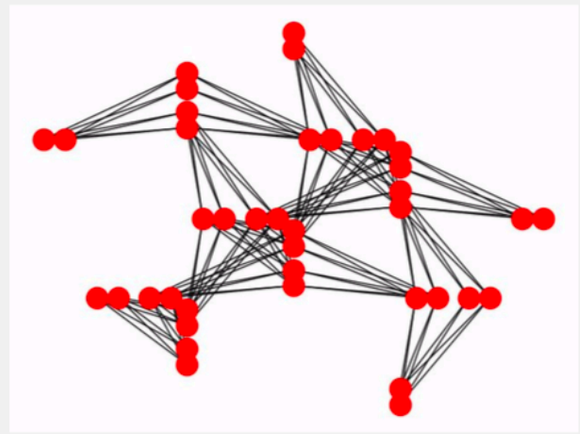
- DWave 2000 Qubit chimera architecture in use.
- Simulated QA can use custom QPU architecture
- improvements from pegasus model
  - larger qbit count - more triplets/anneal
  - higher connectivity

Pegasus:

15 couplings/qbit

~5000 qbits

ETA: 2020



👋 Please have a chat with Parker or Eric! We would like to discuss ideas and ongoing studies.

Thank you for your attention!

Backup