01

Quantum Genetic Algorithm for Continuous Variables Optimization

LAI , CHIA-TSO

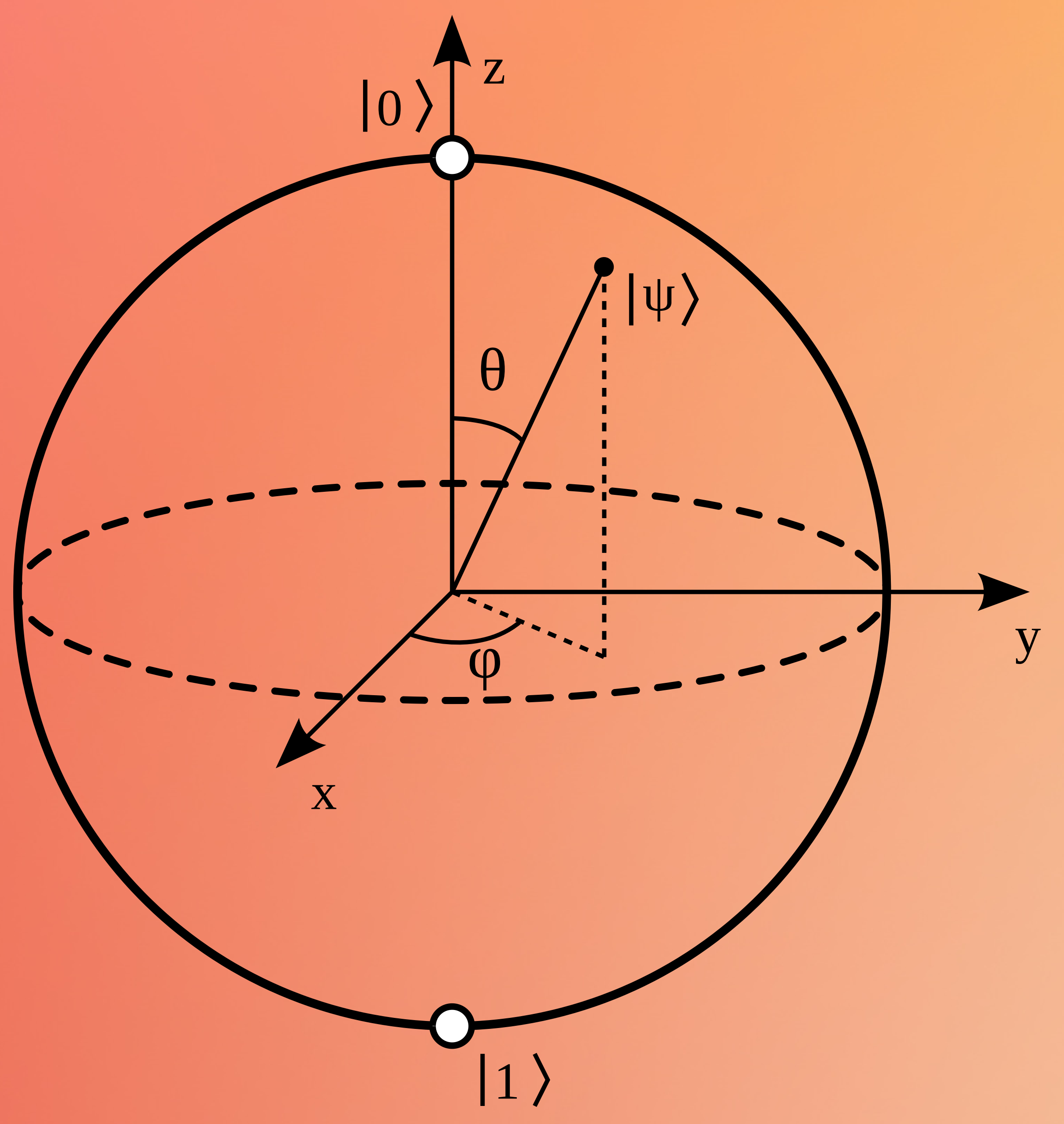
Mapping continuous variables onto Bloch ball

Each qubit has 3 degrees of freedom

Encode continous variables to r,θ, Φ

$\theta \in [0, \pi]$ $\phi \in [0, 2\pi)$ $r \in [0, 1]$

n qubits can encode 3n variables



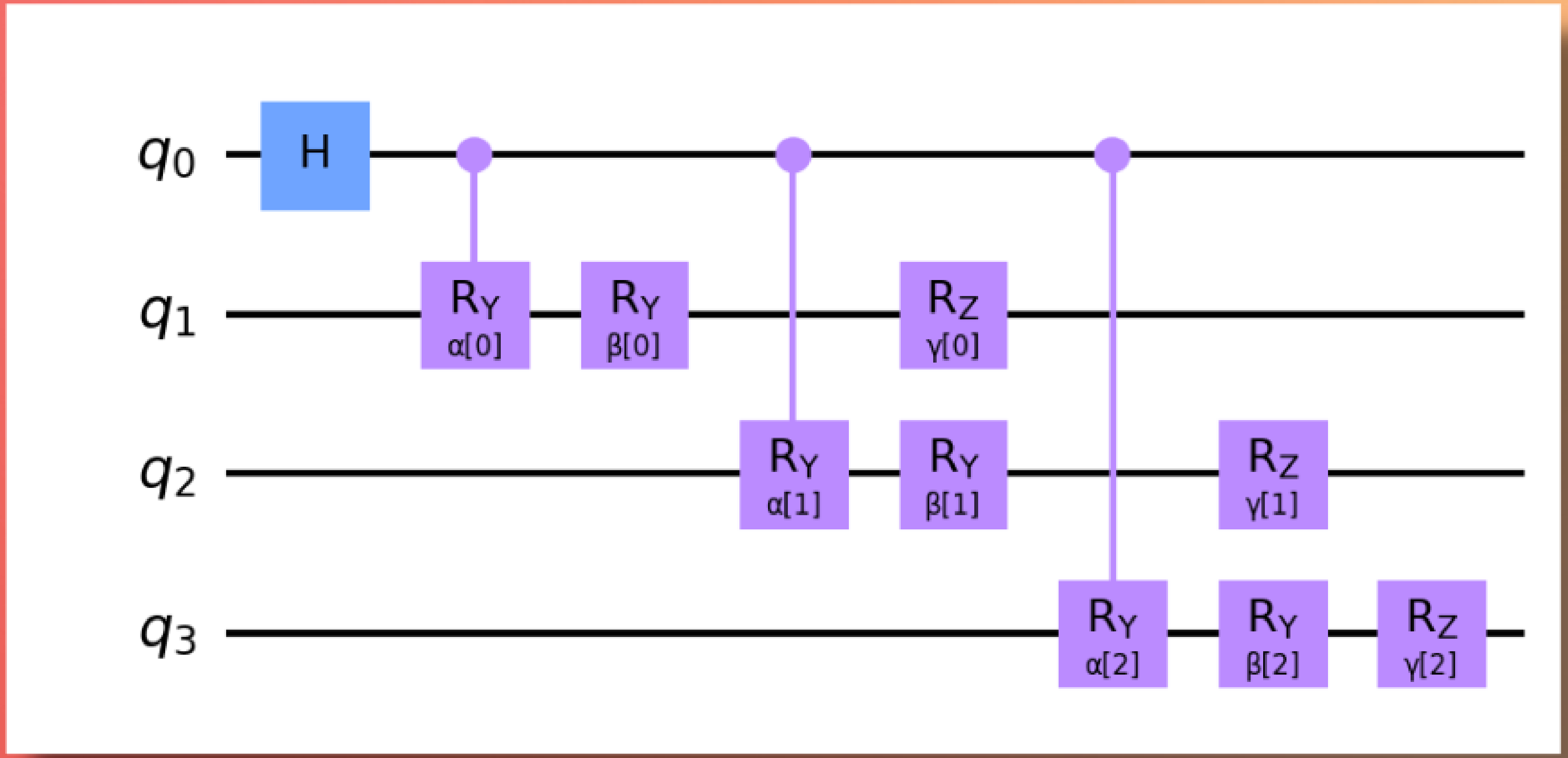
Encoding Scheme

One ancilla qubit for entanglements

C-Ry gates encode r, Ry gates encode θ, Rz gates encode ɸ

Variables' domains are embedded in:

$\theta \in [0, \pi] \quad \phi \in [0, 2\pi) \quad r \in [0, 1]$



i.e. 9 variables are encoded in 3 qubits + 1 ancilla qubit

Single qubit tomography for readout

Density matrix of a single qubit:

$$\rho = \frac{1}{2}(\mathbb{I} + \vec{r} \cdot \vec{\sigma})$$

with the Bloch's vector r :

$$\vec{r} = r(\sin\theta\cos\phi, \sin\theta\sin\phi, \cos\theta)$$

Single qubit tomography can reconstruct the Bloch's vector with measuremt statistics in X,Y,Z basis

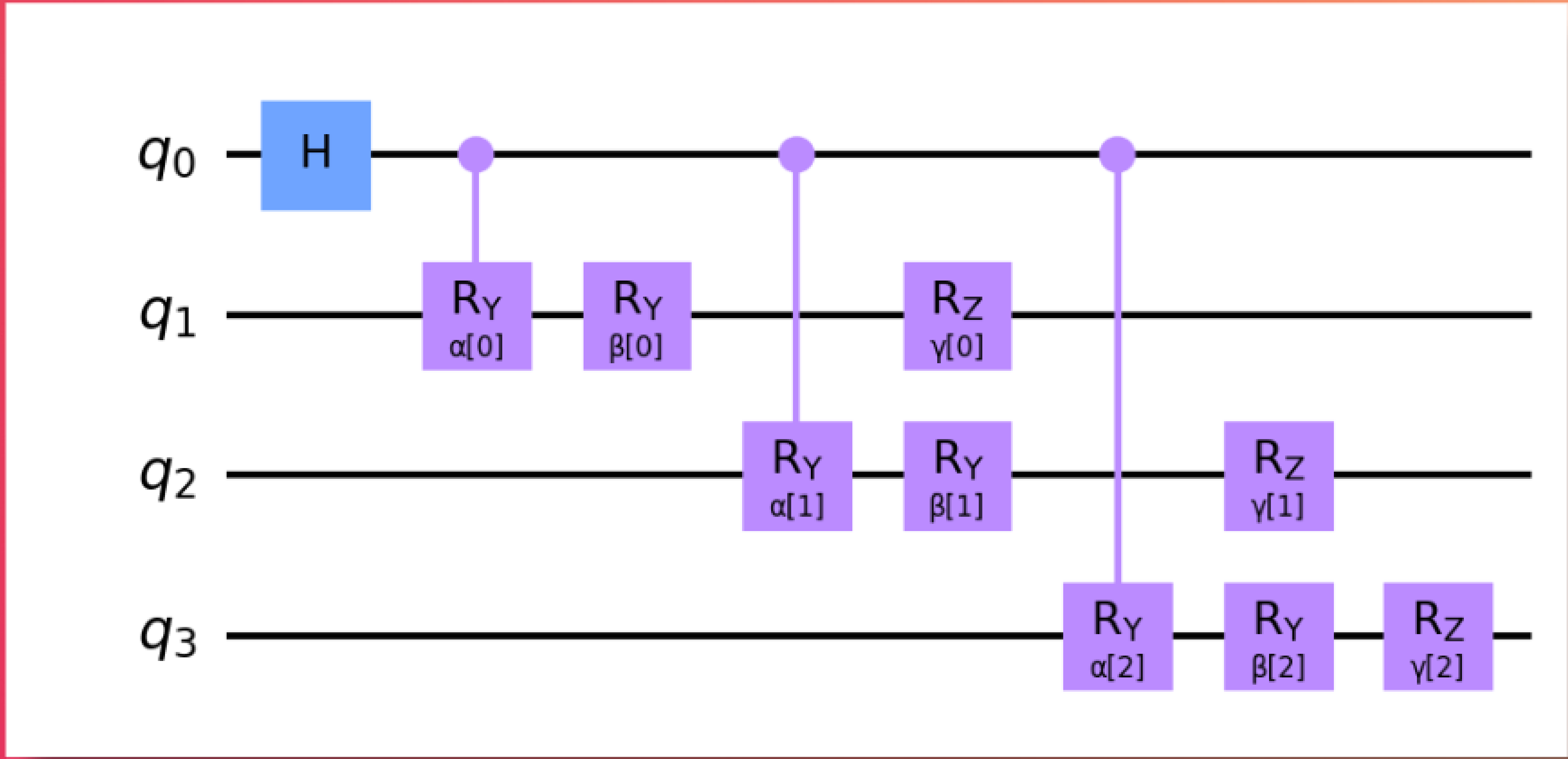
$$r_i = \frac{n_{i+} - n_{i-}}{n_{i+} + n_{i-}}$$ i = x,y,z

r, θ and Φ can then be computed

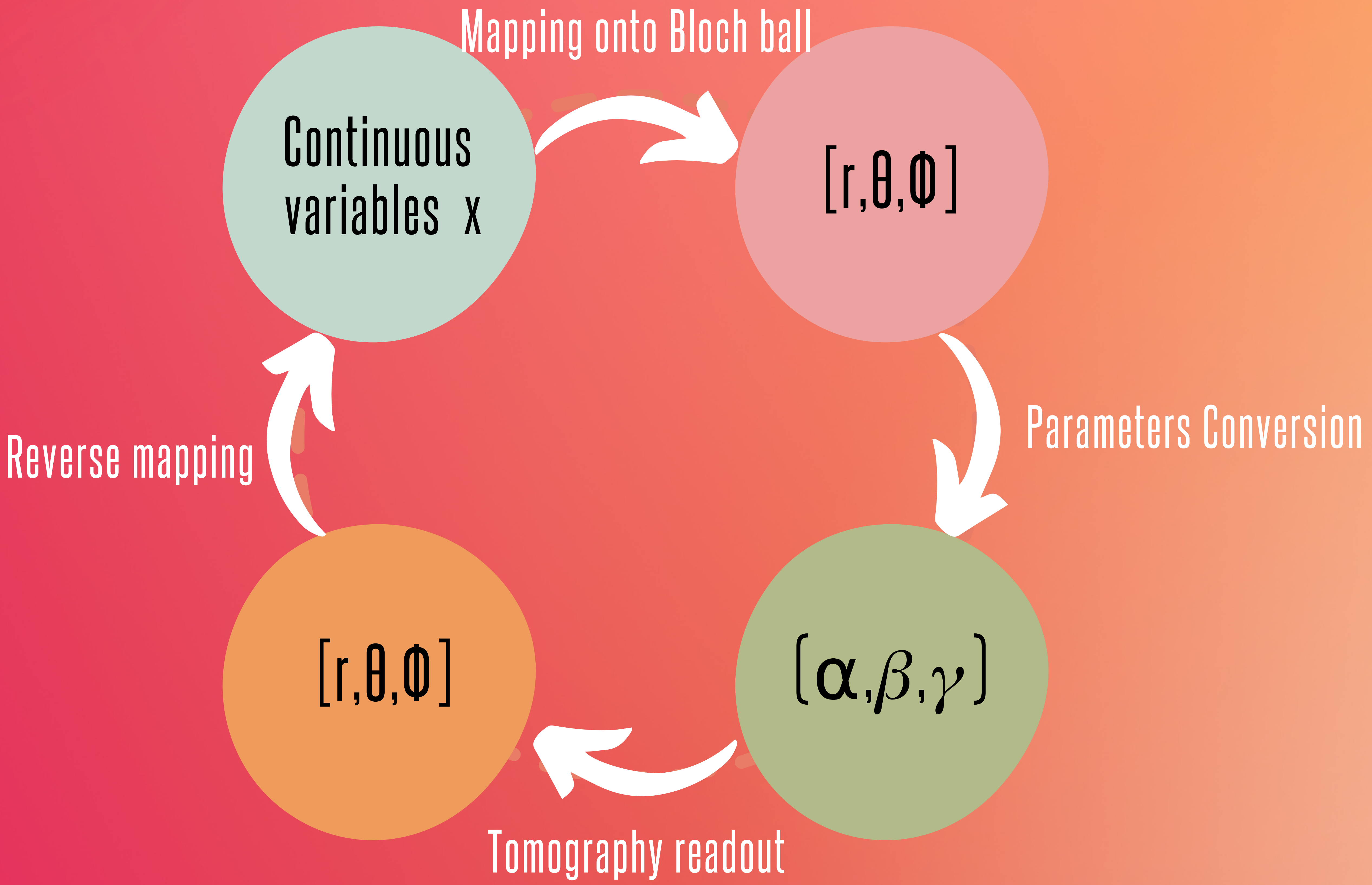
Parameters Conversion

To prepare the qubits in the desired states parametrized by the Bloch's vectors, (r,θ,Φ) needs to be mapped to the corresponding rotation angles (α,β,γ)

$$\alpha = 2\cos^{-1} r \qquad \beta = \sin^{-1}(r \sin\theta \cos\phi + \sqrt{1-r^2}\sqrt{1-\sin^2\theta\cos^2\phi}) \qquad \gamma = -\tan^{-1}(\tan\phi)$$



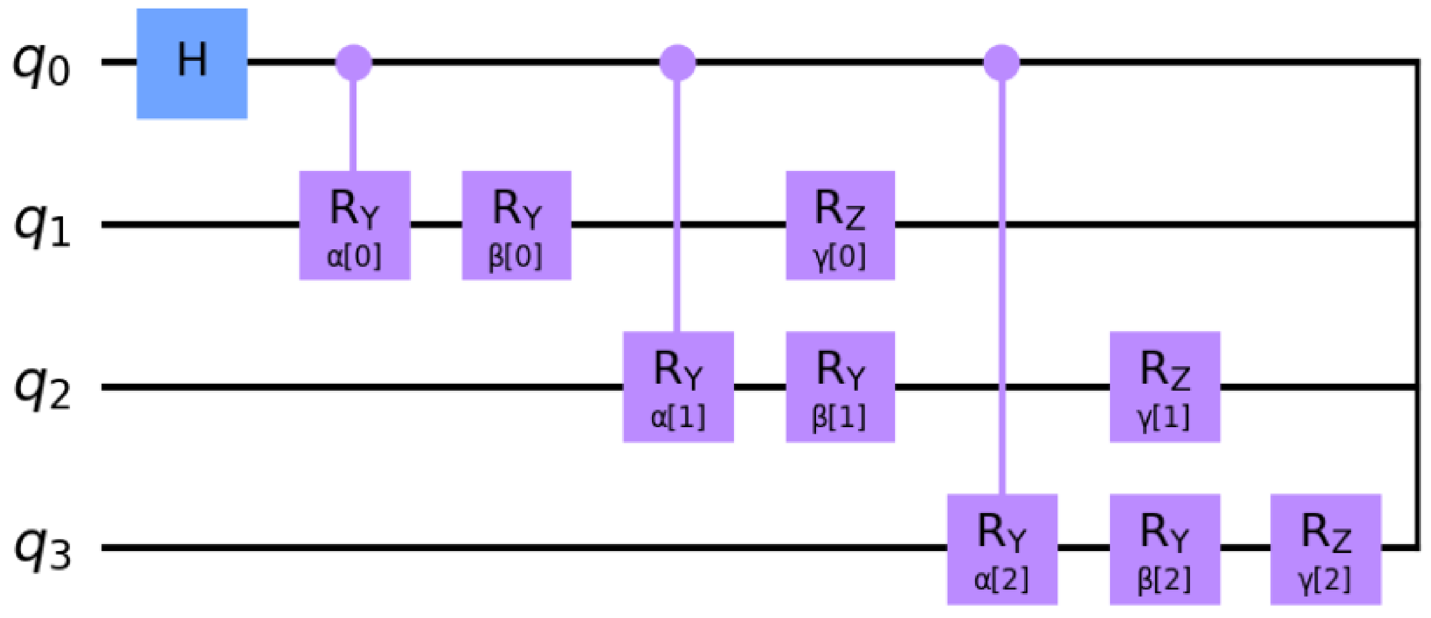
Quantum Subroutine



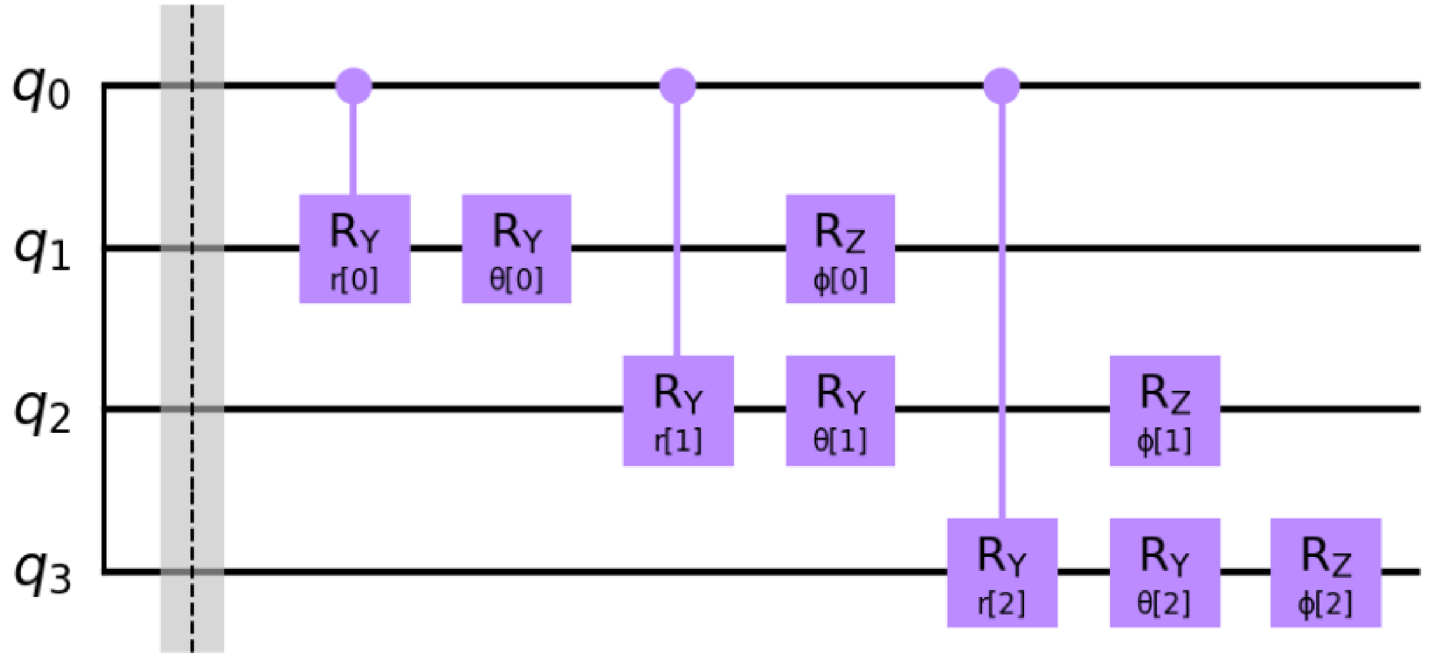
Genetic algorithm for updating variables

1. Map r, θ and Φ back onto continuous variables x
2. Fitness function f(x) is evaluated at values of continuous variables
3. Selection & crossover are carried out as in classical genetic algorithm
4. Map the offspring variables onto Bloch balls again and encode them with C-Ry, Ry & Rz
5. Mutations are implemented via rotating qubits by random angles with C-Ry,Ry & Rz gates (subject to mutation probability)

Quantum Genetic Algorithm



variables encoding+mutation



initial variables

quantum encoding+mutation

single qubit tomography

genetic algorithm

optimal solutions

Example 1 : 4-variable function

Maximize the following function:

$$\sin\left(\frac{x_1}{x_4 \cos(\log(x_1^2 x_2 x_3^{-1}))}\right)$$

$x \in [0,5]$

optimal solutions found:

[1.29926748 0.85551854 0.2270866 1.00295998]
[1.68530558 1.2071522 0.02476588 5.]
[0.62232728 1.01879038 1.17282002 0.87602927]

$f(x) = 0.9999$

Example 2 : 8-variable function

Maximize the following function:

$$-\sum_{i=1}^{8} x_i^2 \qquad x \in [-5,5]$$

optimal solutions found:

[-1.25515279 0.48400317 -0.98176339 -2.12424325 2.39234725 2.00217702 -0.56907336 -0.19341411]

$f(x) = -17.34$

[-0.29110907 -1.24442516 -0.85416384 2.38024127 -2.05833463 -2.25445061 -0.93734387 -0.27380217]

$f(x) = -18.03$

Example 3: 28-variable function

Maximize the following function:

$$
\begin{aligned}
f(\vec{x}) &= \sin\left(x_1 x_2^{-1}\cos\left(\log\left(\frac{x_1^3 x_3}{x_4}\right)\right)\sin\left(\frac{x_5}{x_2}\right)\right. \\
&+ \cos\left(x_6^{1/2} x_1 x_5^{-2}\right) - x_9^2\left(x_{10} - x_{11}x_1 x_4^{-1}\right) \\
&+ \sin\left(\frac{x_7^3}{x_1 x_3 + x_4}\right)\cos\left(x_8 x_3^{-1}\sin(x_7)\right) \\
&+ \cos\left(x_{12}^2 - x_9 x_{10}\right) + \cos\left(\frac{x_{21}x_{22}}{x_{23}} - \sin(x_{24})\right) \\
&+ \cos(x_{13}x_{14})\log\left(\frac{x_{15}}{x_{16}} + x_{14}x_{15}^2\sin\left(x_{13}\cos\left(\frac{x_{16}}{x_{15}}\right)\right)\right) \\
&+ \sin\left(\frac{x_1^2 x_{17}}{x_{18}} + \cos\left(\cos\left(\frac{x_{19}}{x_{20}}\right)\right)\right) \\
&+ \sin\left(x_{25}x_1 x_6^{1/2} x_{26}\right) + \cos\left(x_{27}x_{28}^2\right) \\
&- \left. x_3\log\left(\left(\frac{x_{27}x_{28}}{x_{21}} - \sin(x_5 x_{11})\right)\right)\right).
\end{aligned}
$$

$x \in [0,5]$

optimal solutions found:

[[0.92630697 2.22237034 1.61545033 3.48605848 4.69768121 2.8382311
0.78667815 1.86360844 1.58148594 1.51355759 0.68528887 4.38997128
1.26120089 4.75850603 0.74106166 4.25201994 4.17688018 0.36750301
4.88589449 3.85447375 2.71955284 0.97689529 2.64715572 0.46407807
2.24957522 0.70126019 3.20821907 4.54911479]

f(x) = 0.9999

[3.09122452 3.79056807 3.01015274 0.87003059 0.18445417 3.92574651
2.78012256 2.94273539 2.09202353 1.92227795 4.95877598 4.98905957
0.10438098 4.93755242 4.31061998 0.15493161 0.111583 0.1381607
0.104821 0.12389866 1.44977122 1.40731733 0.26787925 2.9834343
2.658288 3.46661219 2.60010425 1.31831125]

f(x) = 0.9999

Example 4 : 4-Variable LLC Circuit Optimization

Objecive: Make the black-box function as close to 200 as possible

Maximize the following function:

$$-(LLC(L1, L2, c1, f) - 200)^2$$

L_1 $\in$ [2.e-6, 200.e-6]
L_2 $\in$ [20.e-6, 500.e-6]
C_1 $\in$ [1.e-9, 10.e-9]
f $\in$ [100_000, 600_000]

optimal solutions found:

[1.31268783e-04 2.25336851e-04 2.66535780e-09 4.89312111e+05] f(x) = -0.003

[2.15860006e-05 1.42802437e-04 8.81499275e-09 2.18828068e+05] f(x) = -0.023

[1.01695929e-04 1.34853641e-04 3.09654886e-09 5.44735499e+05] f(x) = -0.012

Summary & Future work

1. The algorithm performs rather well when there are multiple solutions in the search space regardless of the number of continuous variables

2. If there is only one or few solutions within the search space (i.e. $\sin(x)$ with $x \in [0,2\pi]$) the algorithm is more unstable and returns less than optimal solutions

3. Initial variables generation, tomography accuracy, selection rate, mutation probability, crossover rules and termination criteria are all parameters that are sensitive and play crucial roles in the performance of global or local search

4. Can the encoding scheme provide any quantum advantage or be proven hard for classical computers to simulate?

Reference

Bermejo, P., & Orus, R. (2022, October 6). Variational quantum continuous optimization: A cornerstone of quantum mathematical analysis. arXiv.org. https://arxiv.org/abs/2210.03136