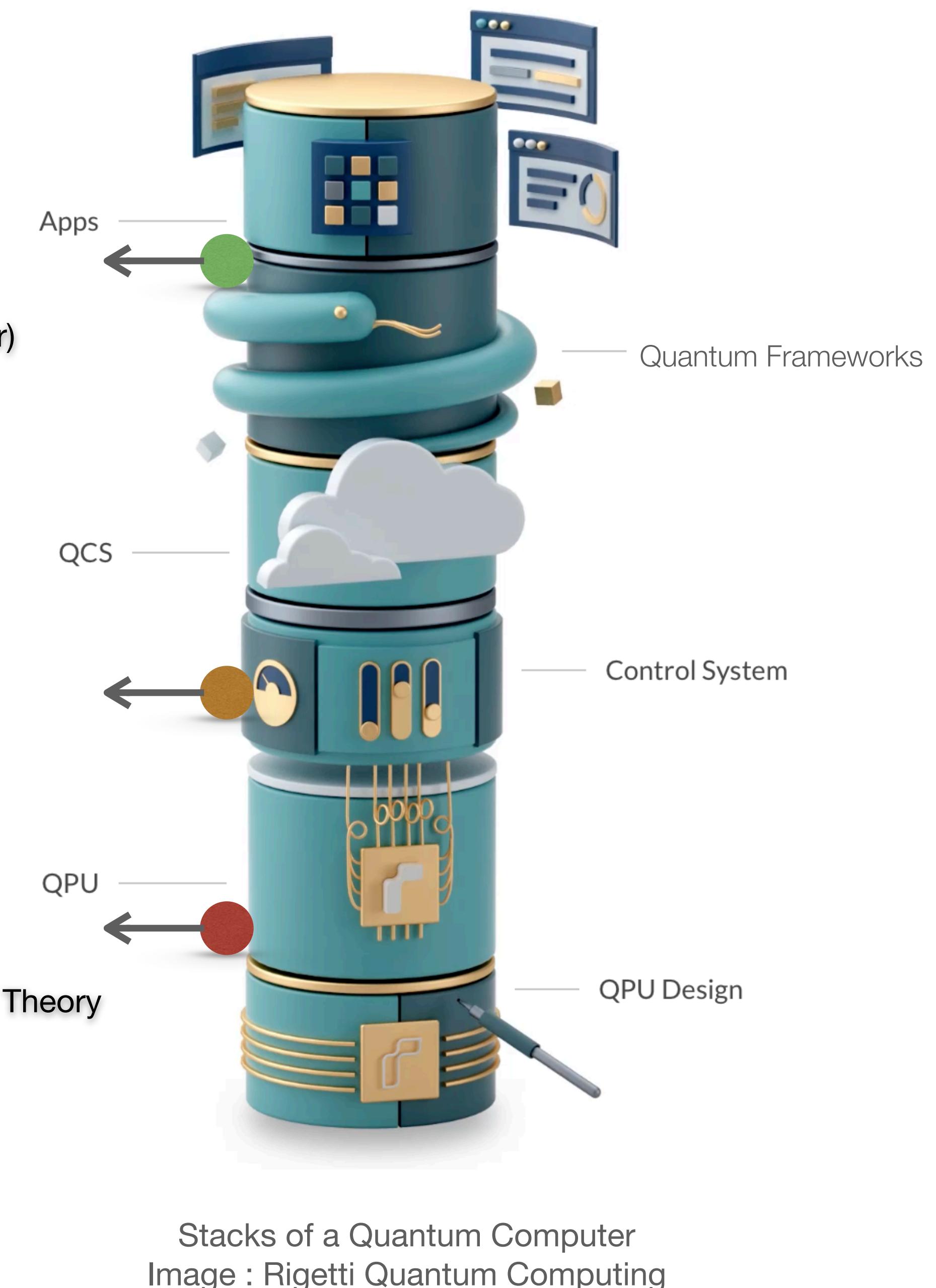
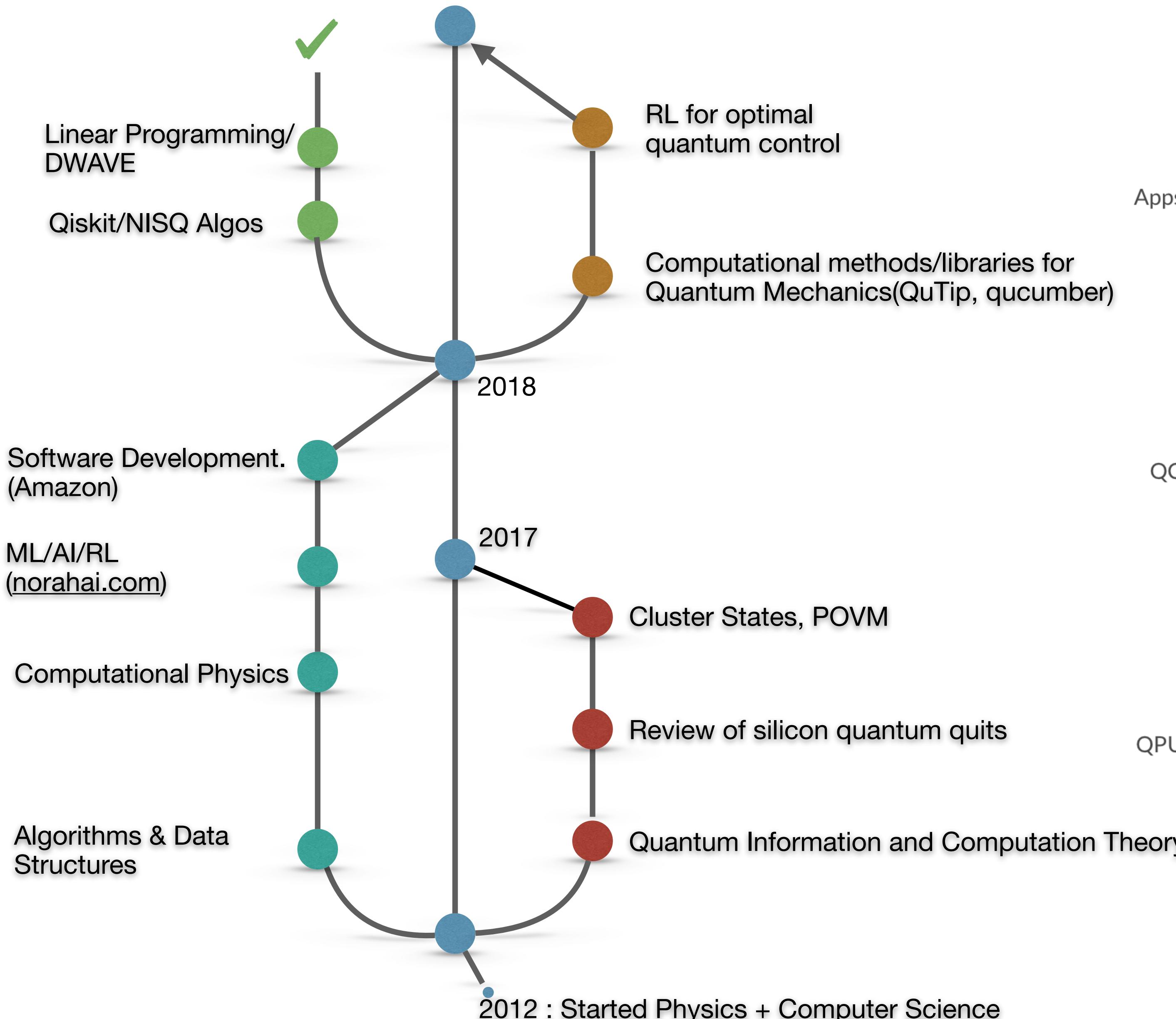


OPTIMAL QUANTUM CONTROL

Kevin Abraham



Hardware Components of a Quantum Computer

Error source ?		
Quantum Data Plane		
Qubit Implementation Error (relaxation, dephasing)		

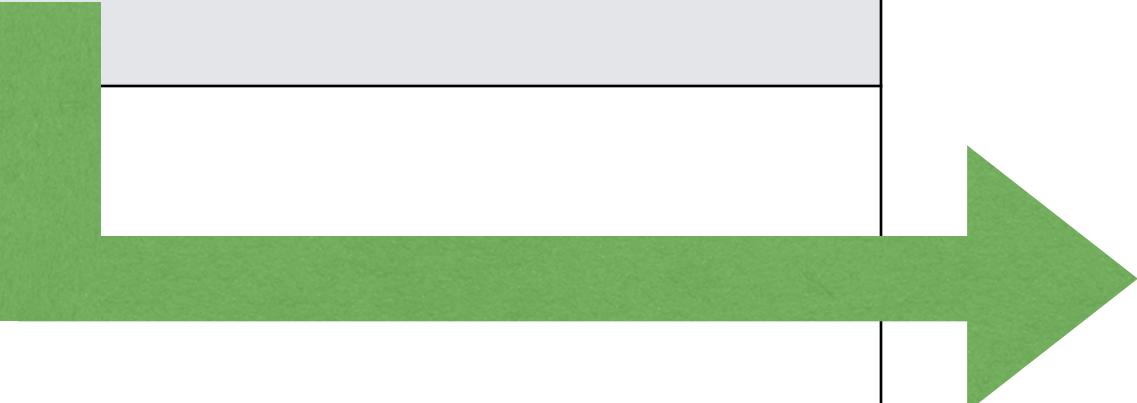
Hardware Components of a Quantum Computer

Error source ?	
Quantum Data Plane	Control and Measurement Plane
Qubit Implementation Error (relaxation, dephasing)	Signal Cross Talk Error

Hardware Components of a Quantum Computer

Error source ?		
Quantum Data Plane	Control and Measurement Plane	
Qubit Implementation Error (relaxation, dephasing)	Signal Cross Talk Error	
Requires extreme quantum engineering and fabrication for improving these		

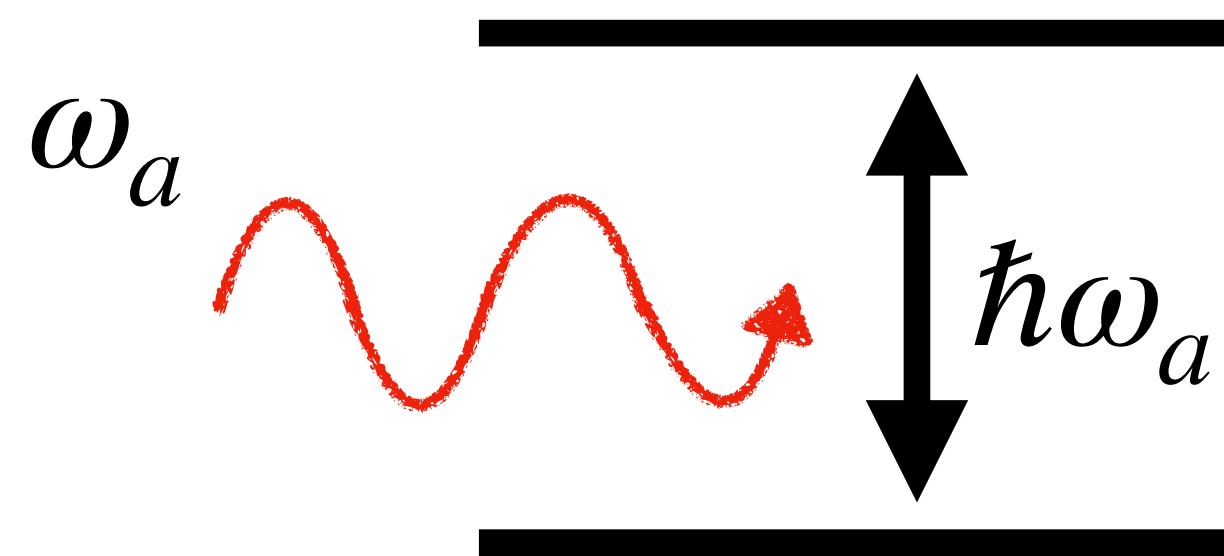
Hardware Components of a Quantum Computer

Error source ?	Reducing the hardware error ?	
Quantum Data Plane	Control and Measurement Plane	Control Processor Plane
Qubit Implementation Error (relaxation, dephasing)	Signal Cross Talk Error	What can the control process do to minimise the impact of noise ?
Requires extreme quantum engineering and fabrication for improving these		

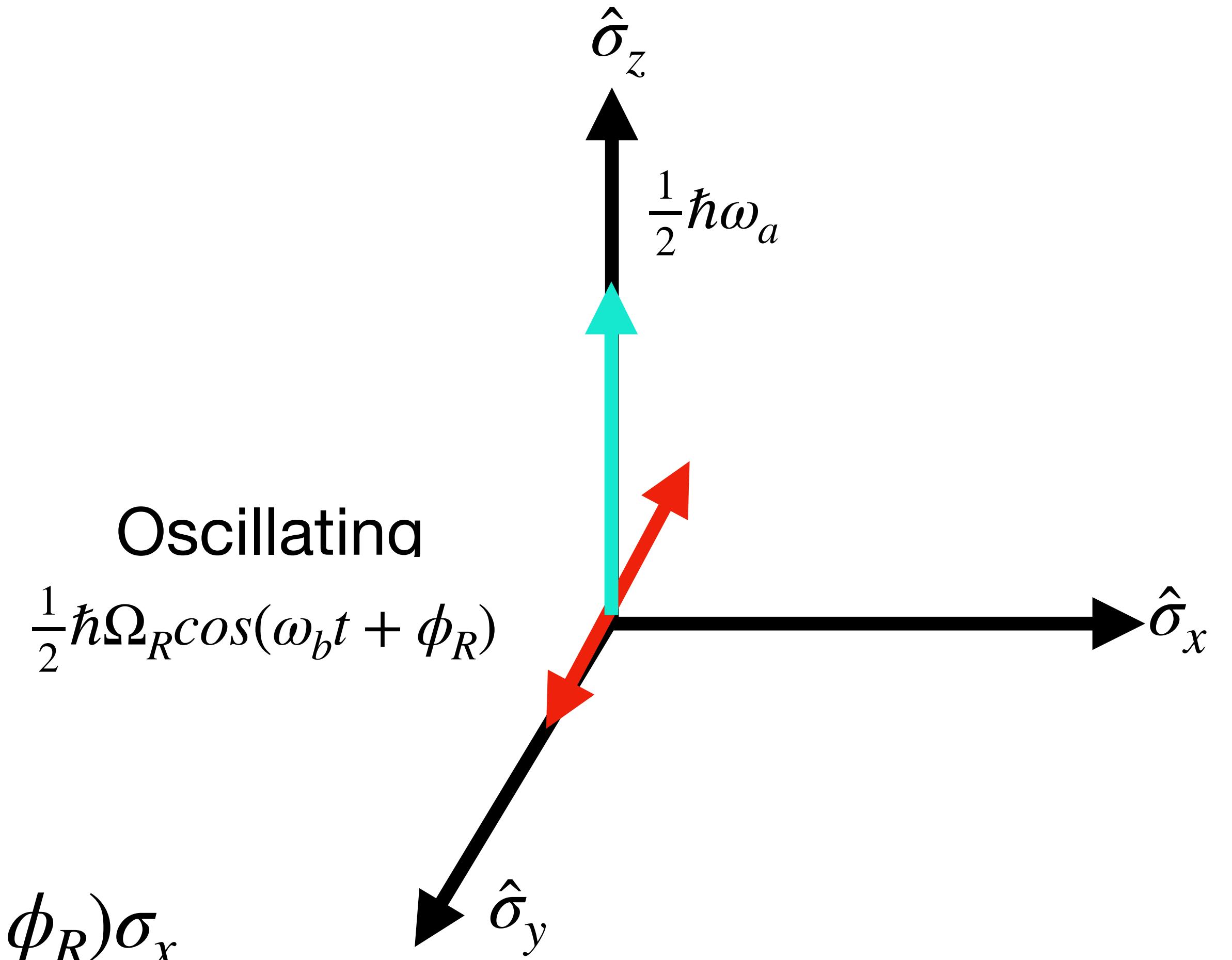


Pulse Shaping

Preface : Rabi Oscillations



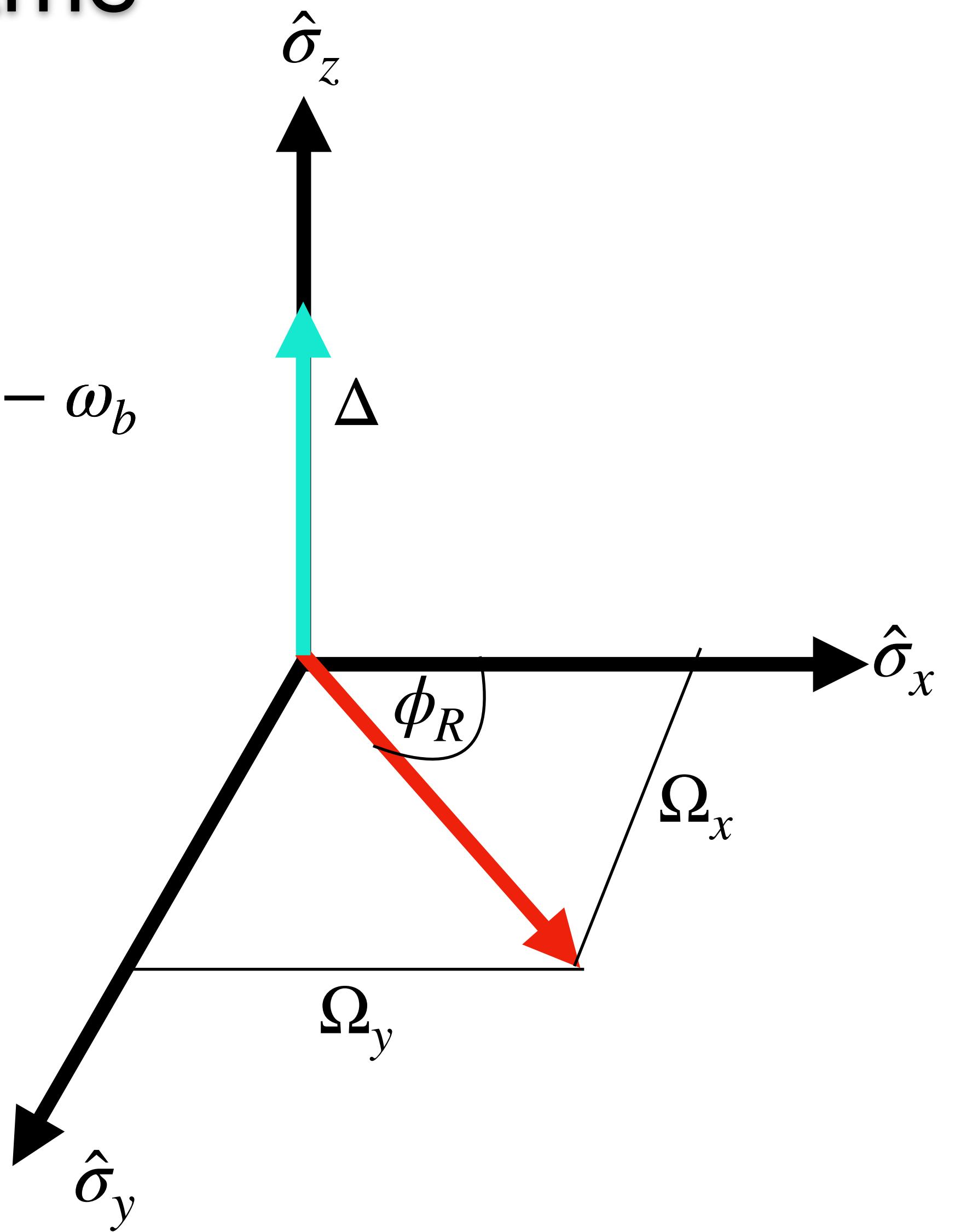
$$H = \frac{1}{2}\hbar\omega_a\sigma_z + \frac{1}{2}\hbar\Omega_R\cos(\omega_b t + \phi_R)\sigma_x$$



Rotating Frame

$$H' = \frac{1}{2}\hbar(\Omega_x\sigma_x + \Omega_y\sigma_y + \Delta\sigma_z)$$

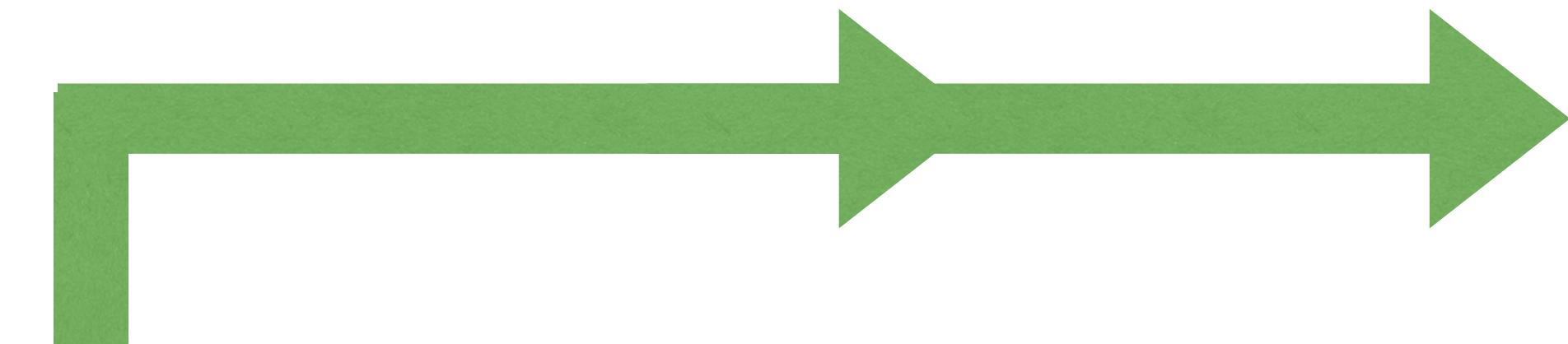
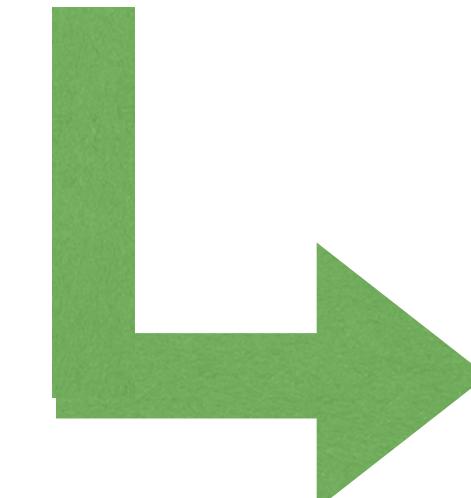
$$\Delta = \omega_a - \omega_b$$



Rabi Pulse

$$H' = \frac{1}{2}\hbar\Omega(X(t)\sigma_x + Y(t)\sigma_y) + \Delta\sigma_z$$

The rotation can be controlled



Rotation Speed \longleftrightarrow Amplitude

Rotation Axis \longleftrightarrow Phase : $\tan(X/Y)$

Rotation Angle = Speed*Duration

= Area under the pulse envelope



RL FOR OPTIMAL QUANTUM CONTROL

What Reinforcement Learning?

Why Reinforcement Learning?

How Reinforcement Learning?

Problem Parameters

$$\Omega$$

Rabi Frequency

$$T$$

Process Time

$$\frac{\Delta}{\Omega}$$

Relative Detuning

$$T_\pi$$

$$\frac{1}{2\Omega}$$

$$\gamma_1$$

Spin Relaxation Factor

$$\gamma_2$$

Decoherence Factor

Problem Statement

Given no information about the system besides simulator of the system find the optimal pulse sequence to perform a unitary operation in the exact T given.

Target: The learned policy should be adaptable to:

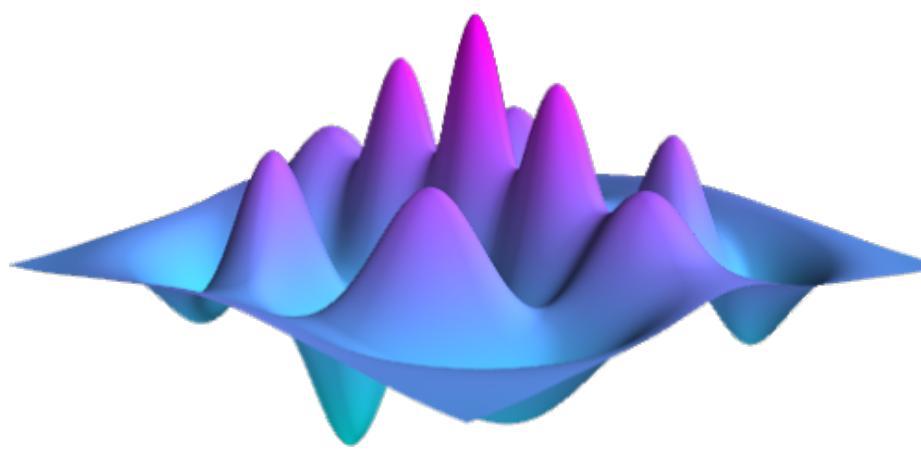
- Variations in the problem parameter
- Different unitary operations

Training

Observables	$\langle \sigma_x \rangle$ $\langle \sigma_y \rangle$ $\langle \sigma_z \rangle$ γ_1 γ_2 F
Reward	$F_{new} - F_{old}$ For end state F_{new} if $F_{new} > 0.99$

$$F \implies F(\Psi_\rho, \Psi_p) = ||\langle \Psi_\rho | \Psi_p \rangle||^2$$

Environment



Qubit dynamics Simulation in Qutip,
with continuous actions



Tensorflow environment
(TFPyEnvironment) compatible

Agent

<https://medium.com/@thechrisyoon/deriving-policy-gradients-and-implementing-reinforce-f887949>

<https://lilianweng.github.io/lil-log/2018/04/08/policy-gradient-algorithms.html>

Neural Network (Actor network)

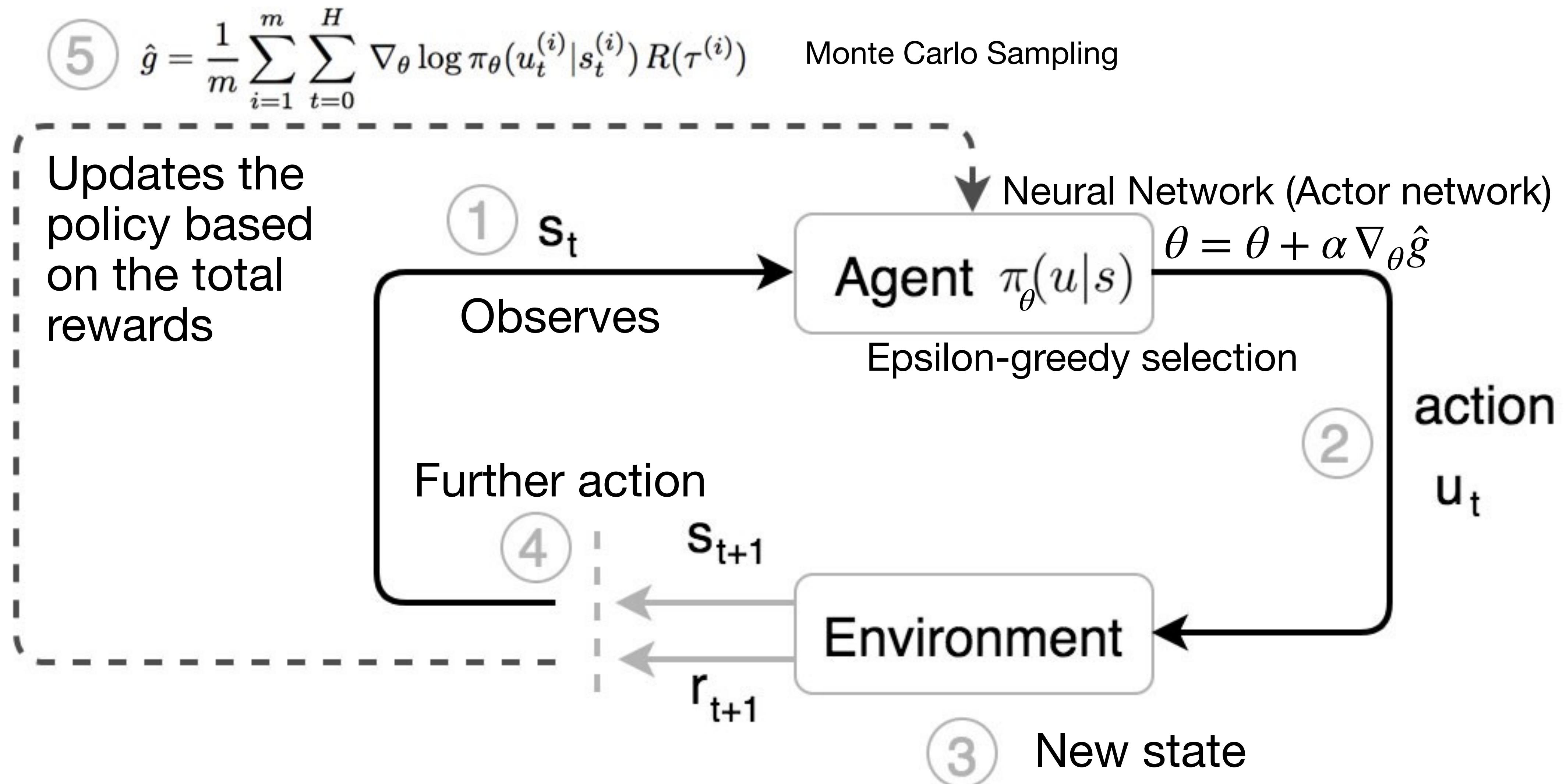
<https://math.stackexchange.com/questions/3179912/policy-gradient-reinforcement-learning-in-a-discrete-action-space>

Policy Gradient Approach with replay buffer (and discounted reward)

https://ml4a.github.io/ml4a/how_neural_networks_are_trained/

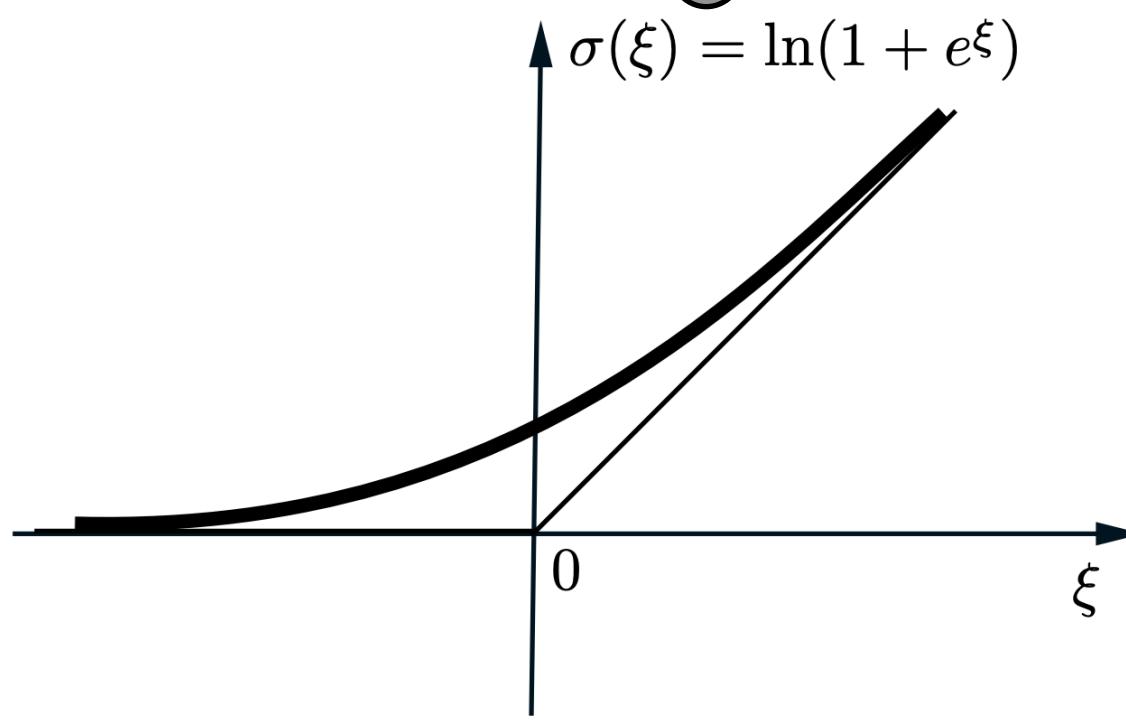
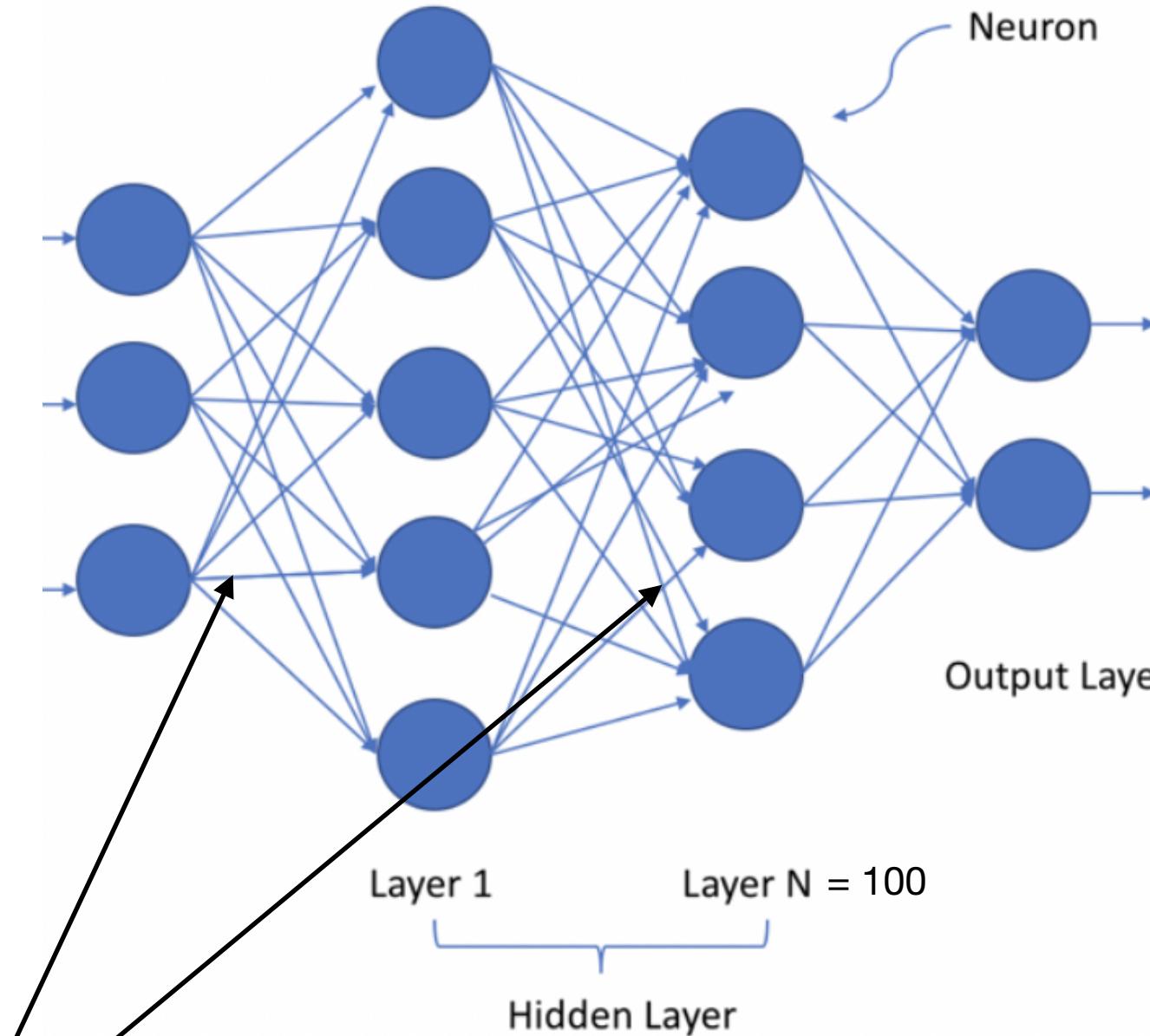
https://medium.com/@jonathan_hui/rl-policy-gradients-explained-9b13b688b146

Policy Gradient

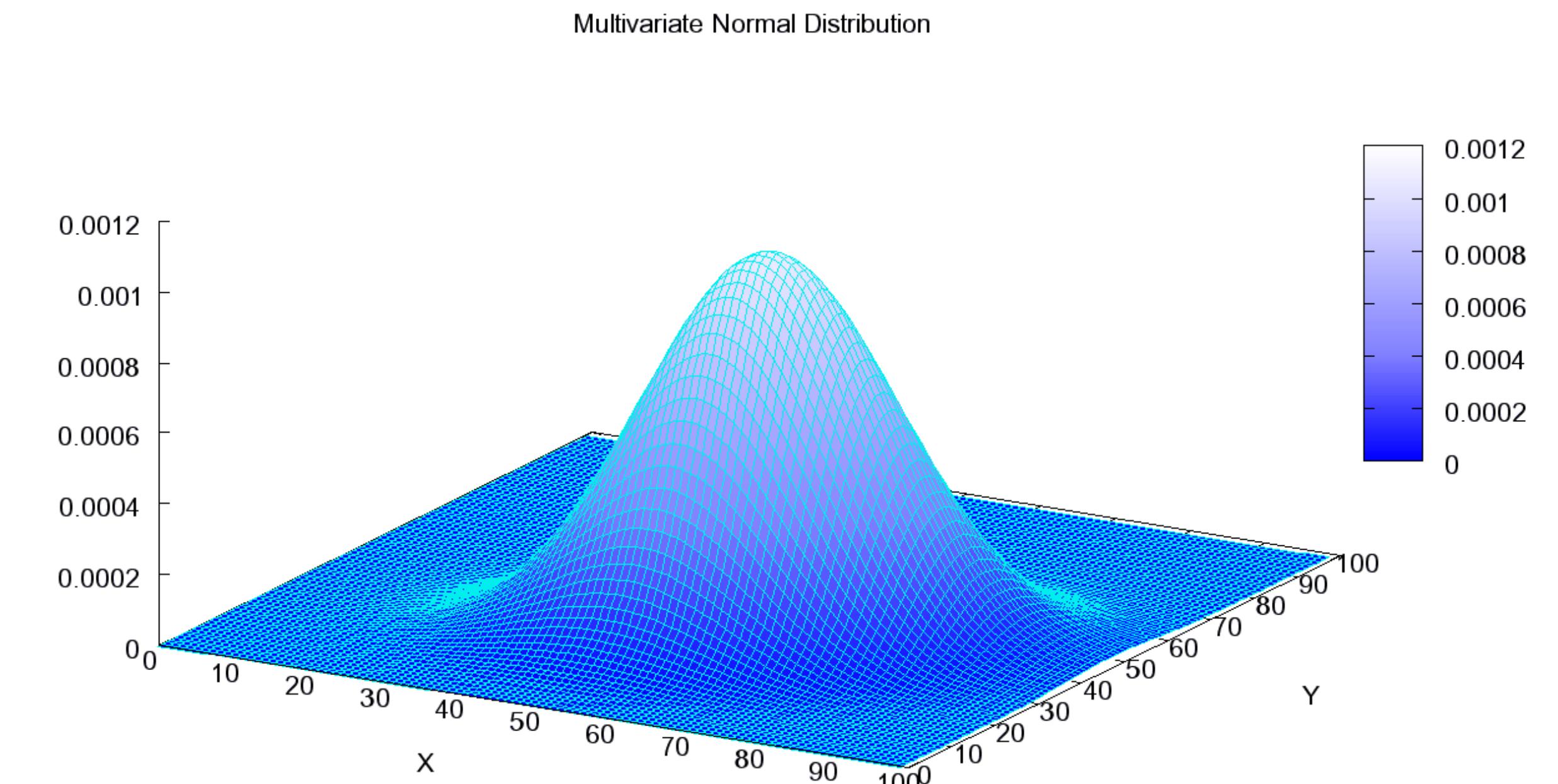


Agent

Observations
Converted to
Batches



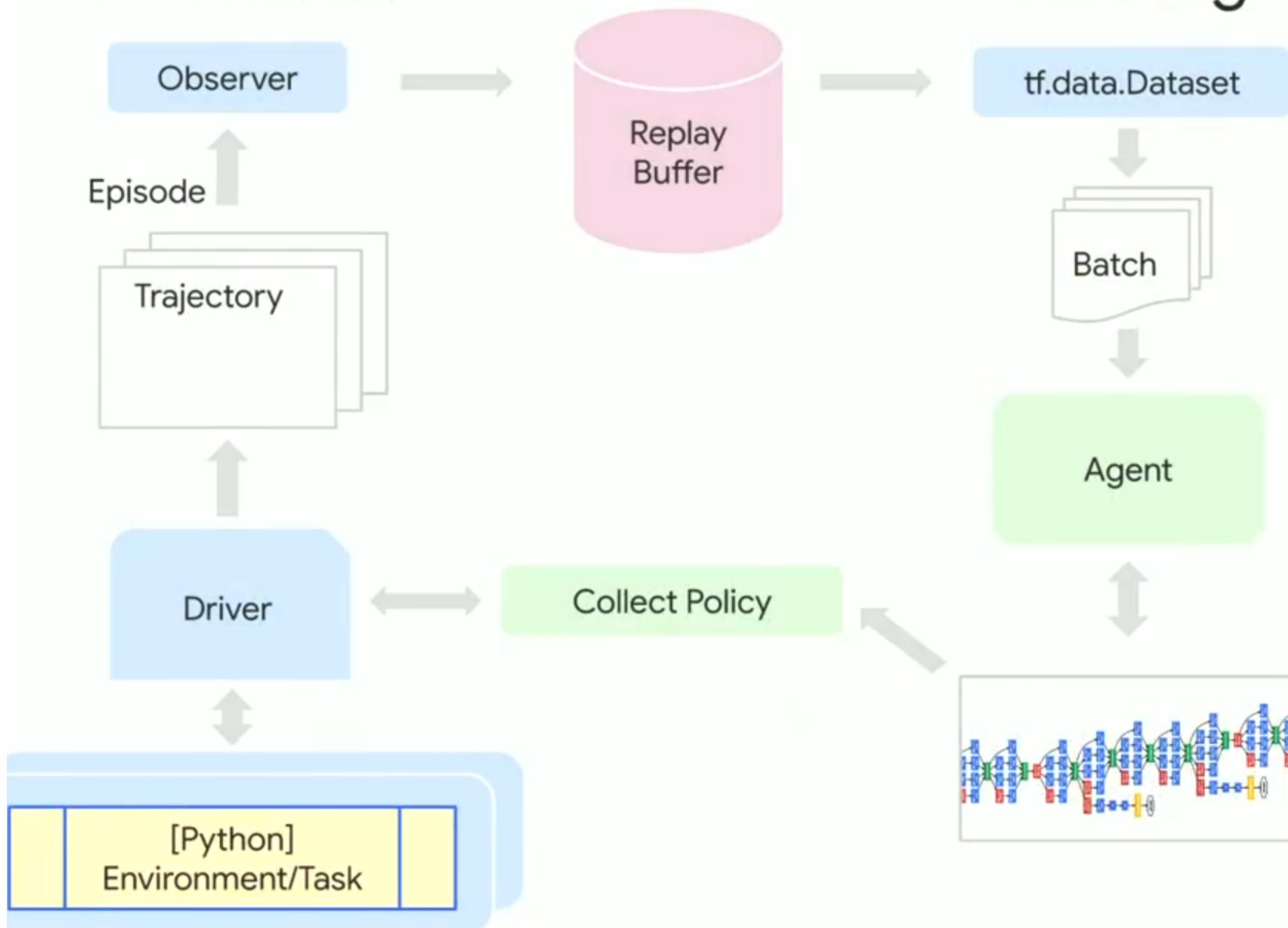
Relu smoothens the fn for differentiability



The softmax function on the a hidden layer gives the probability distribution. The Output layers are the possible set of actions.

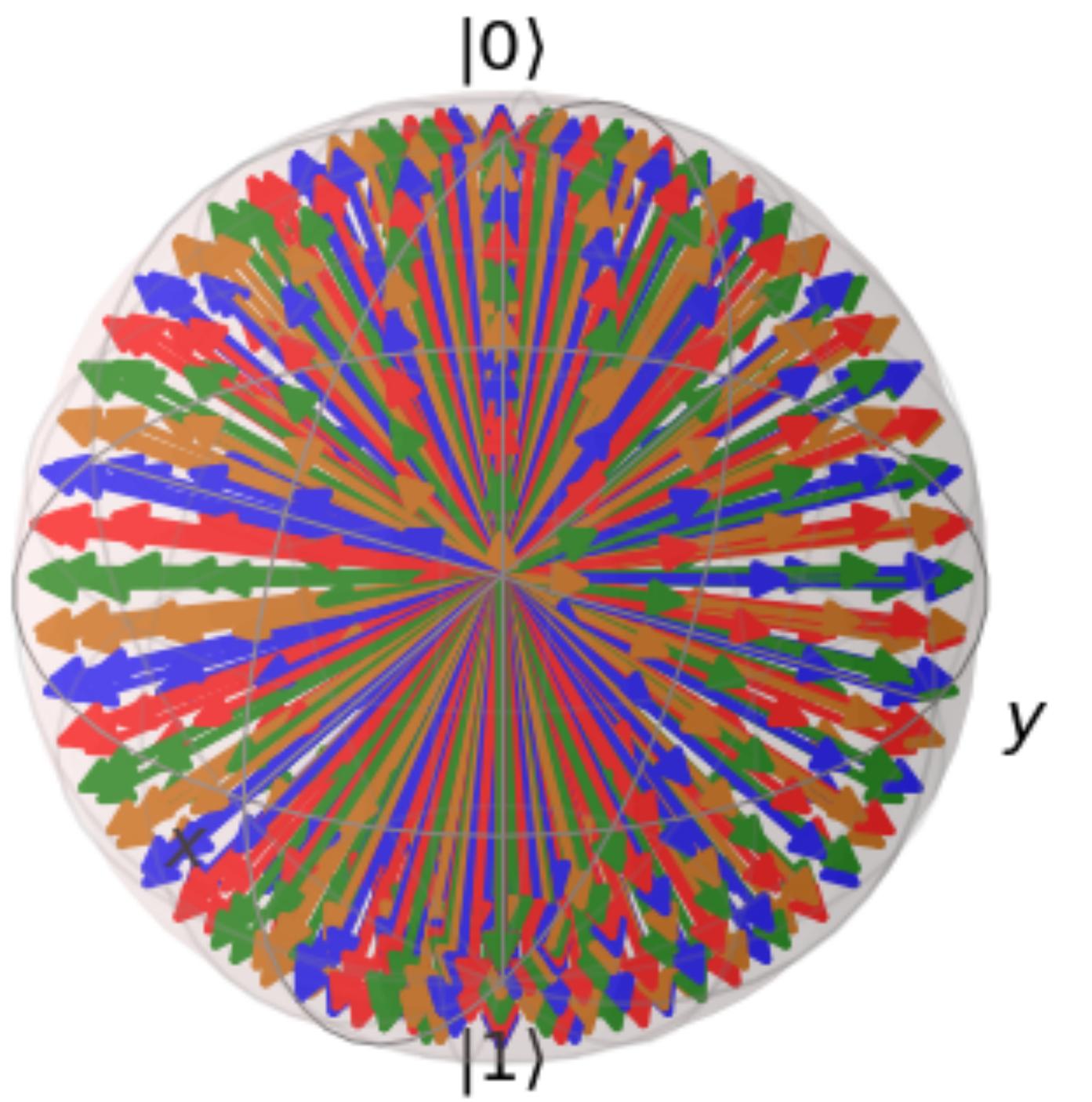
Uses Adam Optimiser for back-propogating errors

Collection



Training

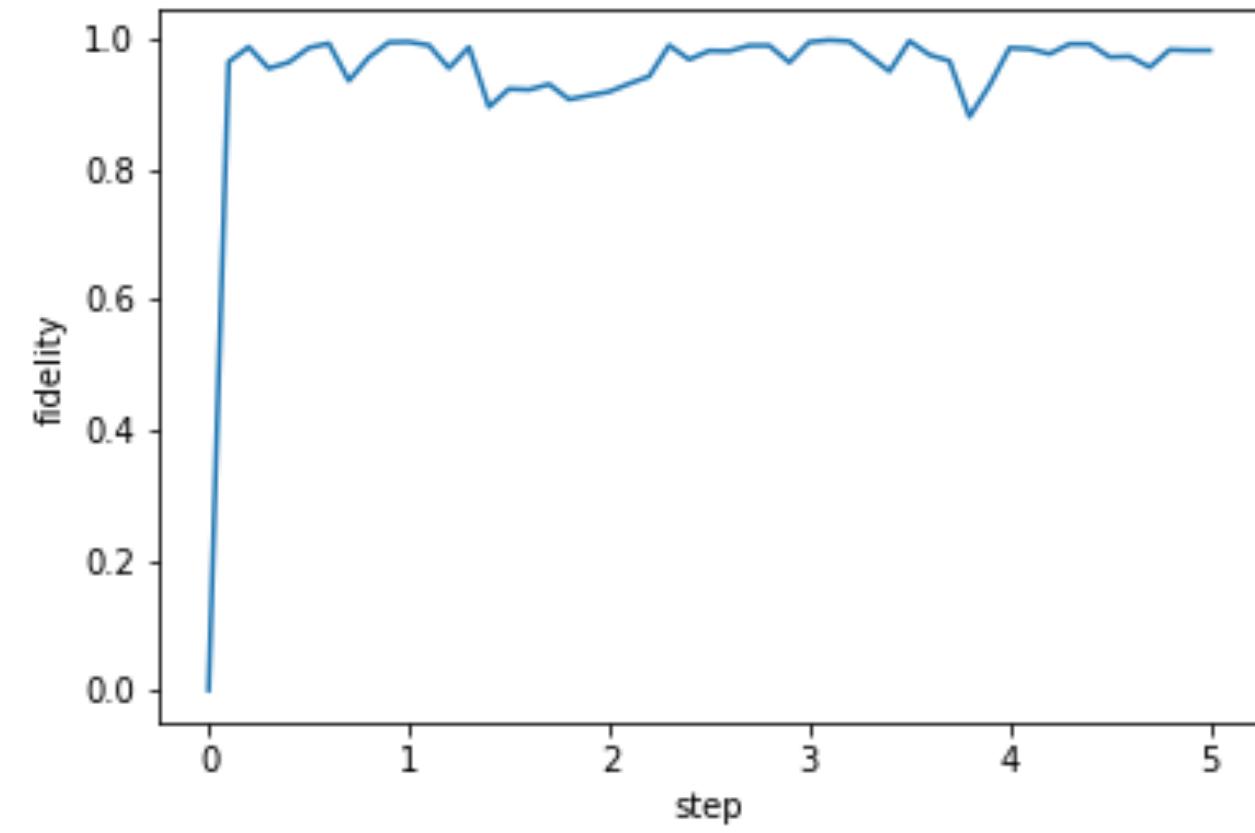
TF-AGENT : Image from TensorFlow Dev Summit 2019



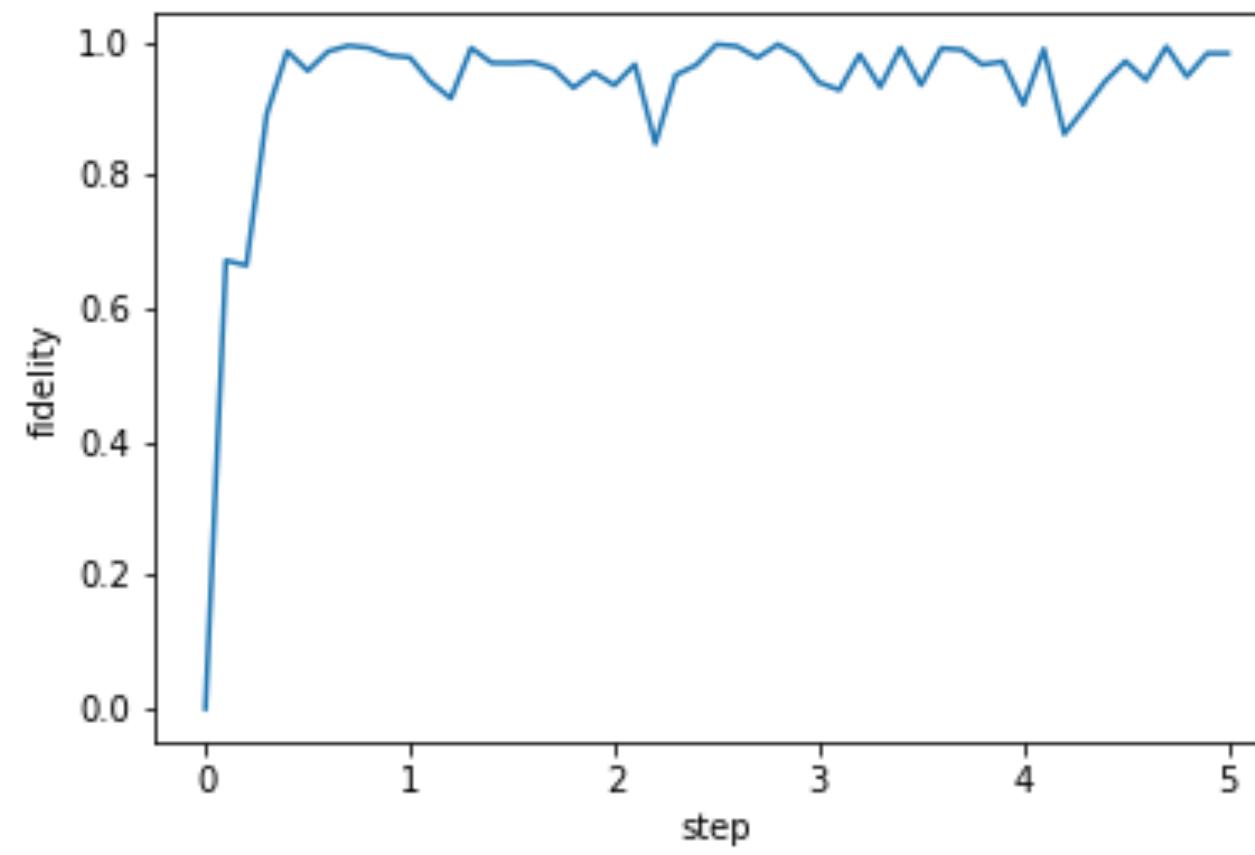
Initial States

Results

$$\Omega = 1 \quad \frac{\Delta}{\Omega} = 0 \quad T = 5 T_\pi$$

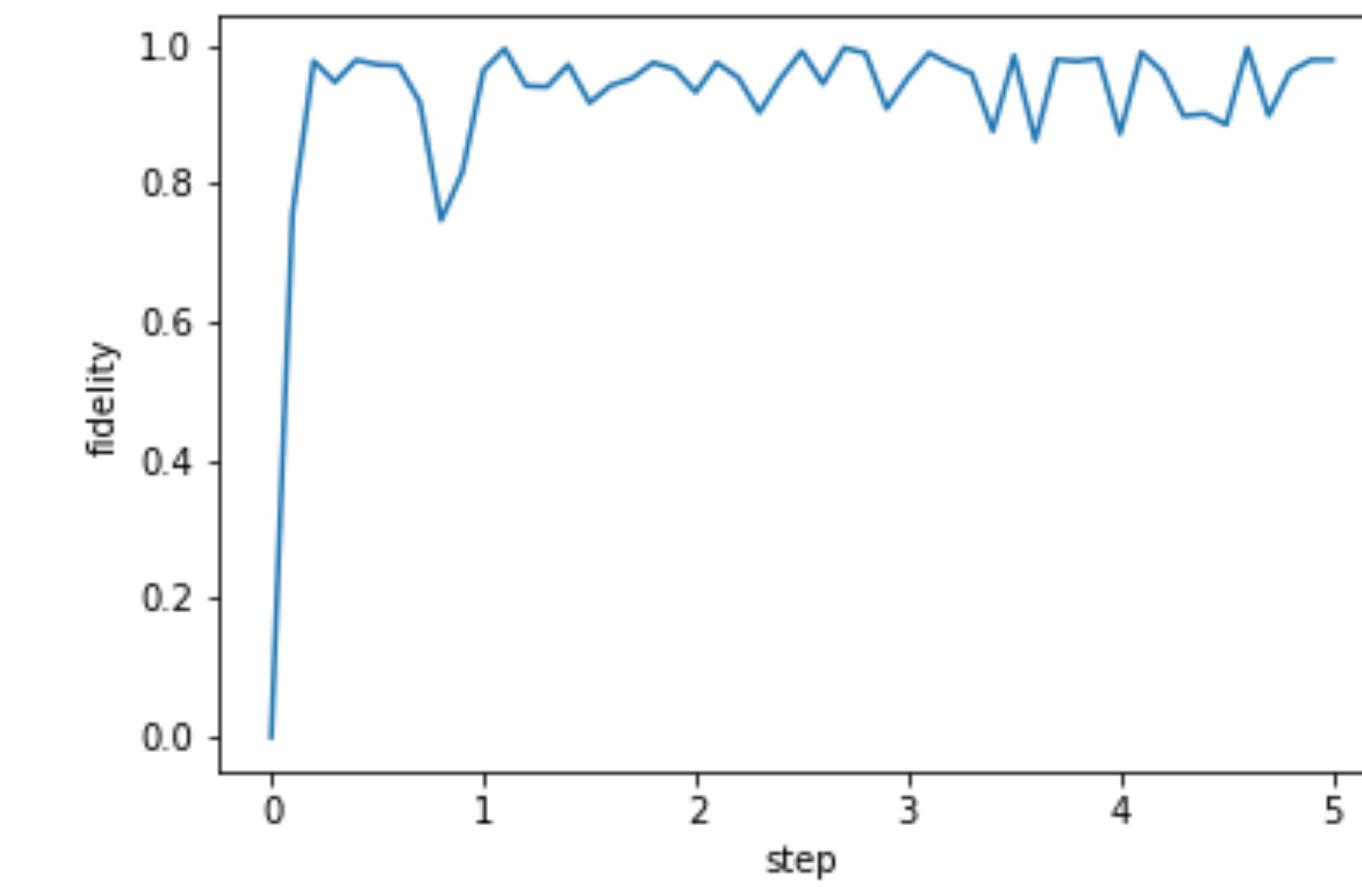


$$\gamma_1 = 0 \quad \gamma_2 = 0$$

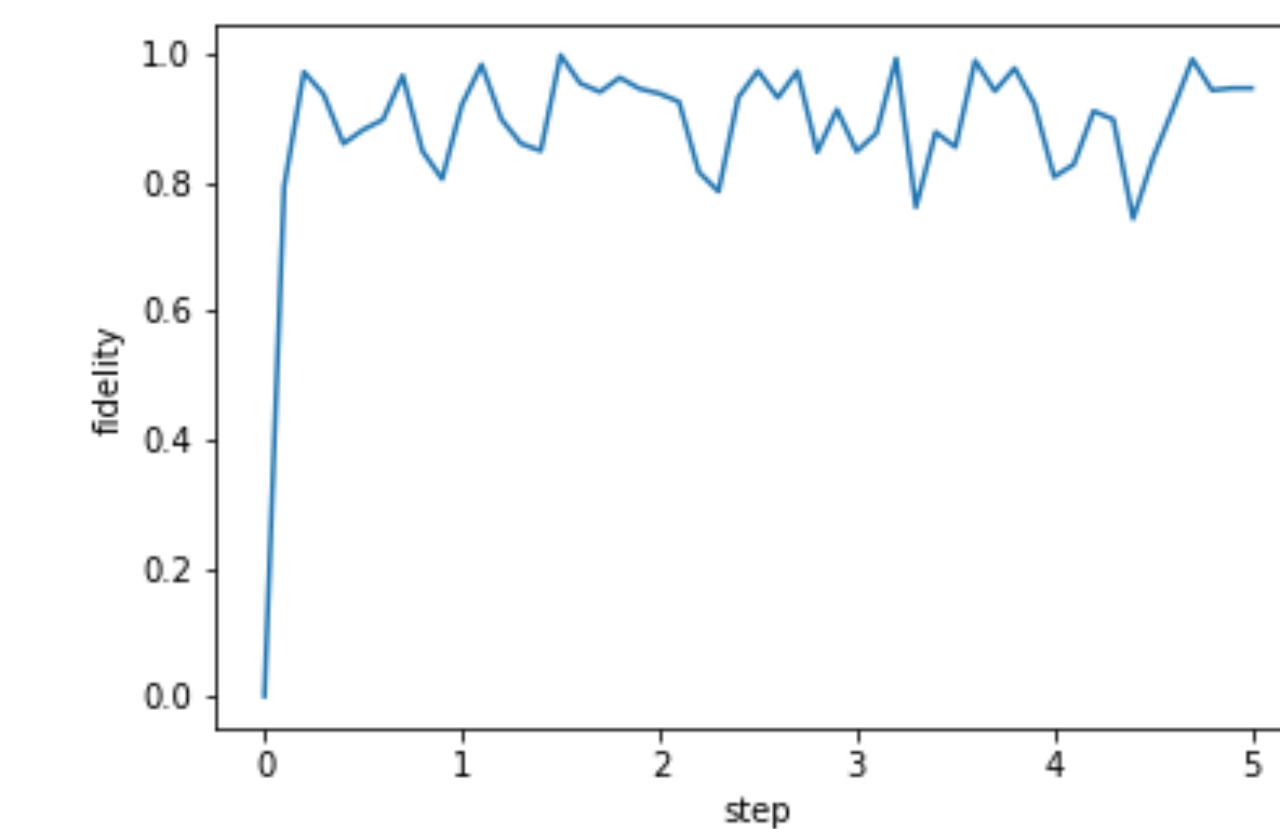


$$\gamma_1 = 0.5 \quad \gamma_2 = 0.5$$

$$\Omega = 1 \quad \frac{\Delta}{\Omega} = 0.3 \quad T = 5 T_\pi$$

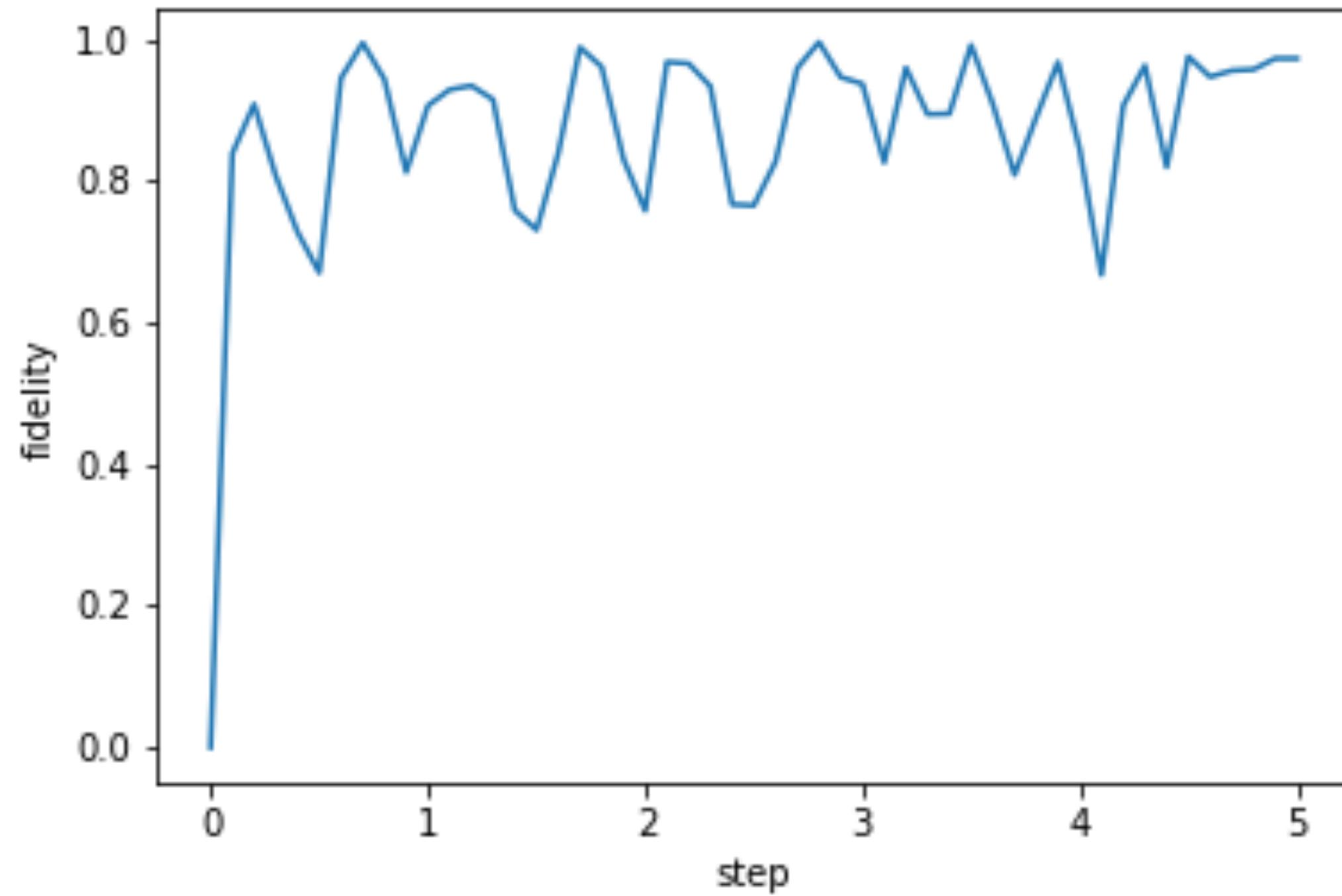


$$\gamma_1 = 0 \quad \gamma_2 = 0$$

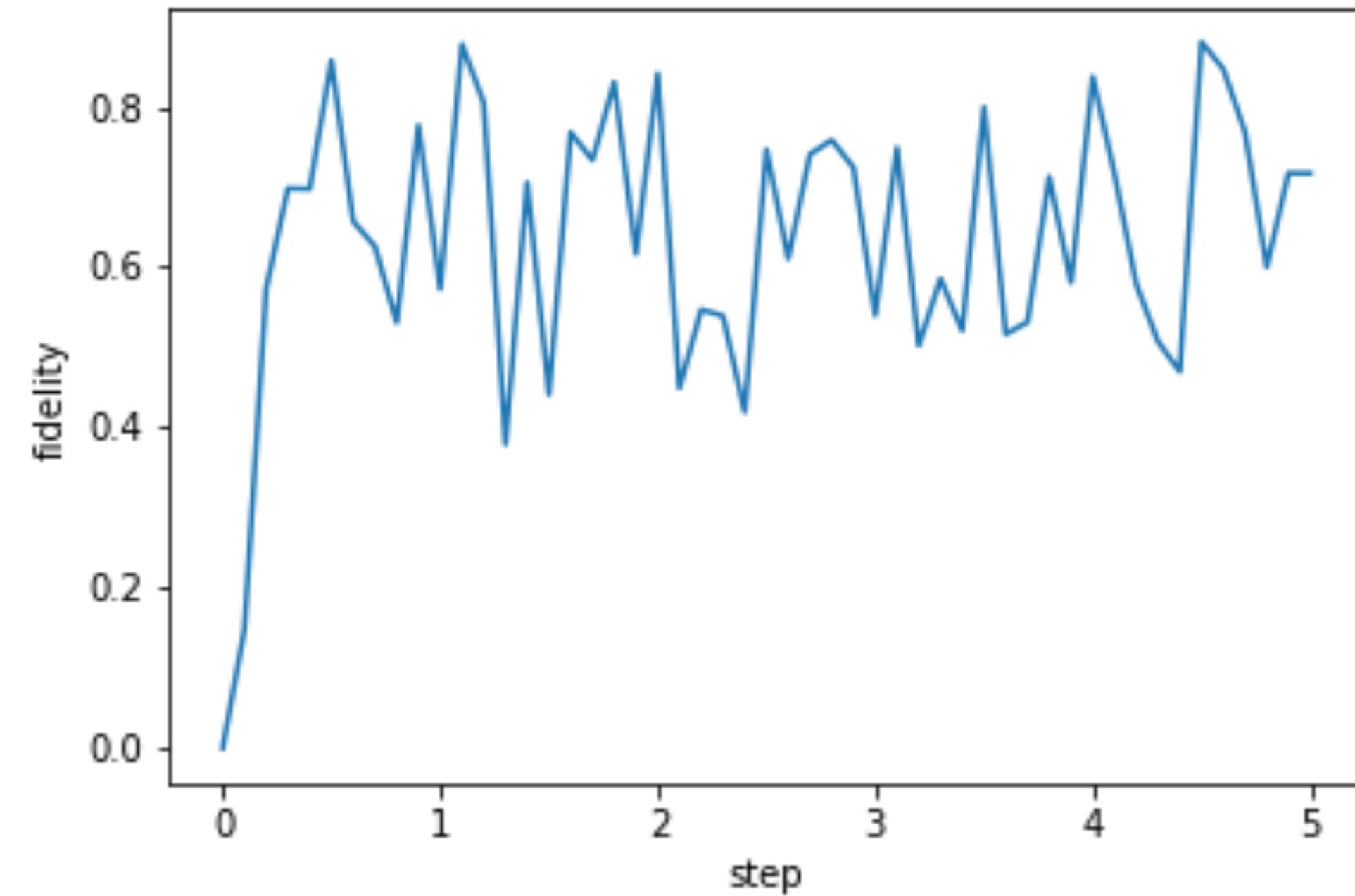


$$\gamma_1 = 0.5 \quad \gamma_2 = 0.5$$

$$\Omega = 1 \quad \frac{\Delta}{\Omega} = 0.7 \quad T = 5 T_\pi$$

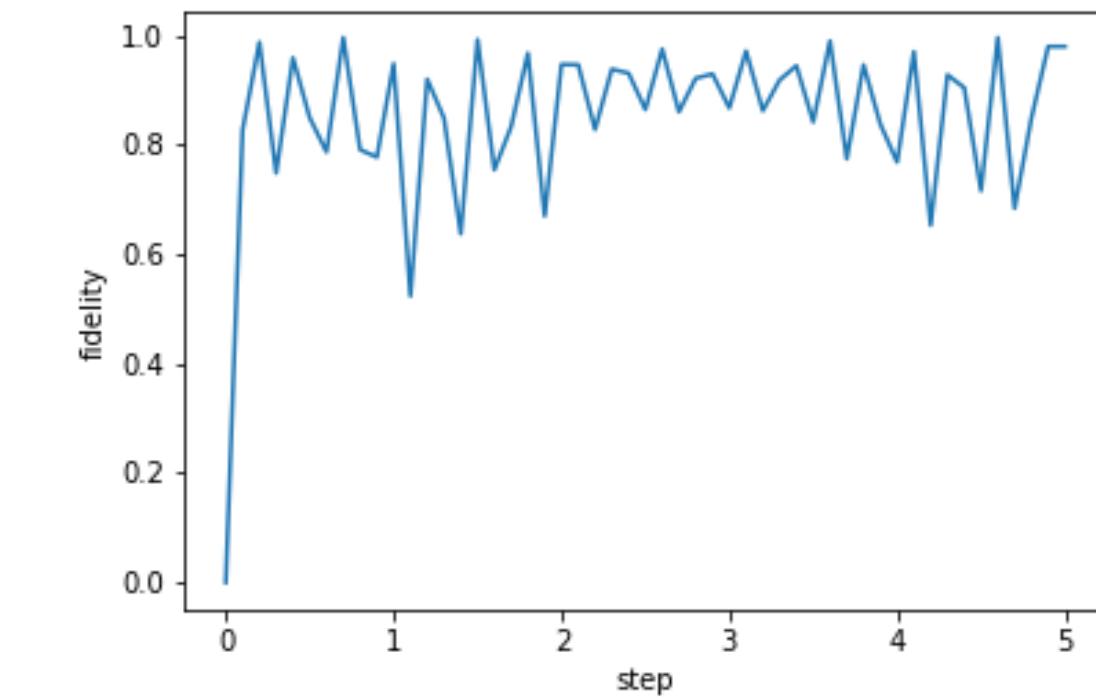
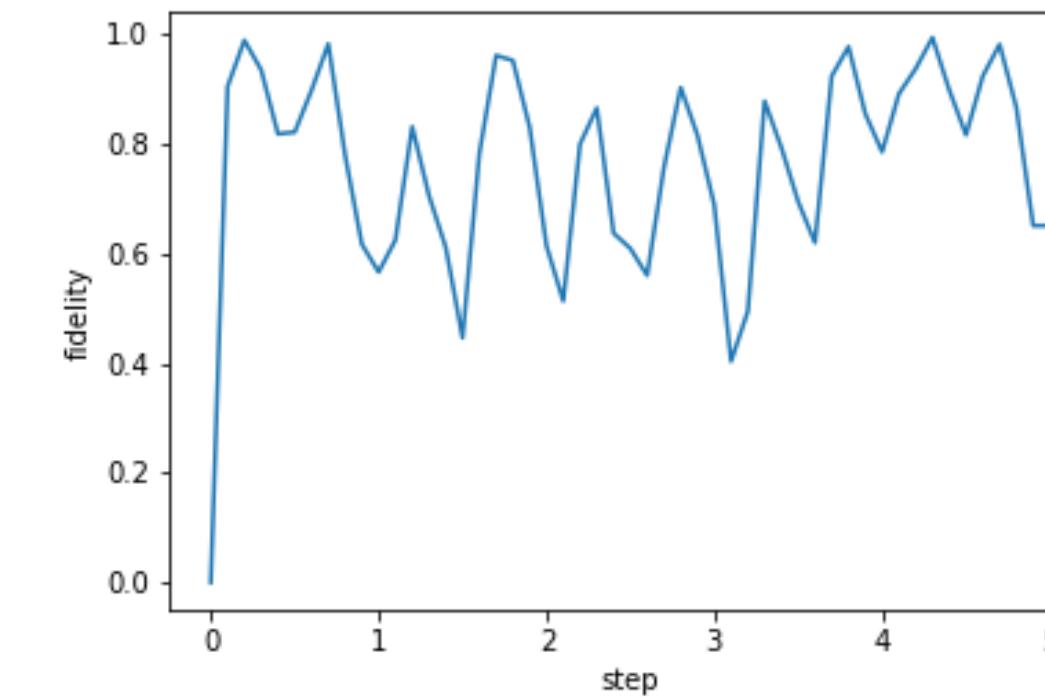
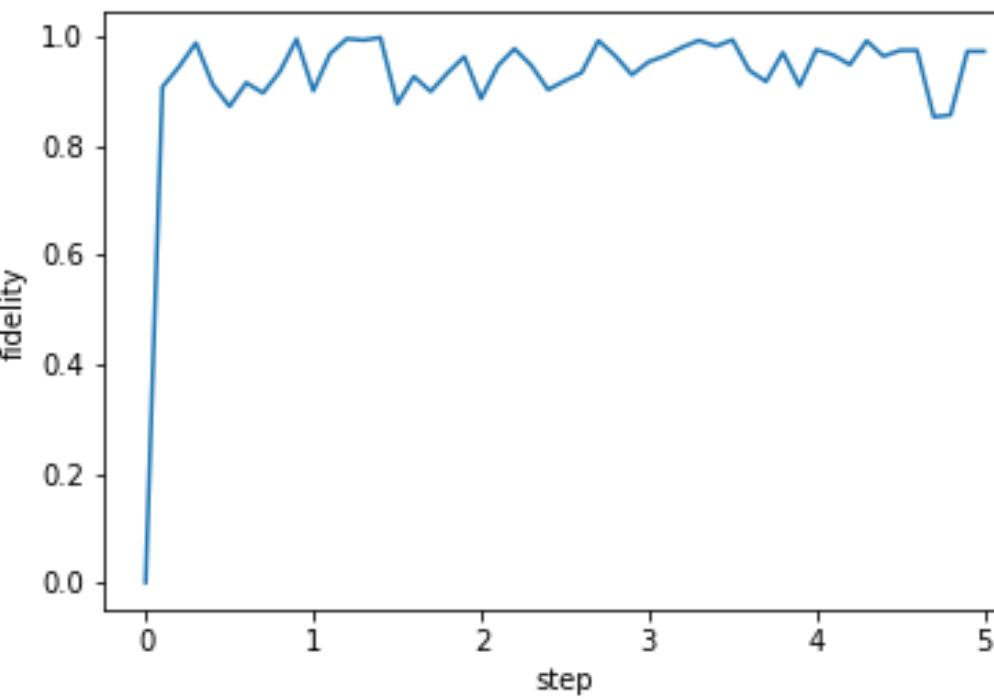


$$\gamma_1 = 0 \quad \gamma_2 = 0.1$$



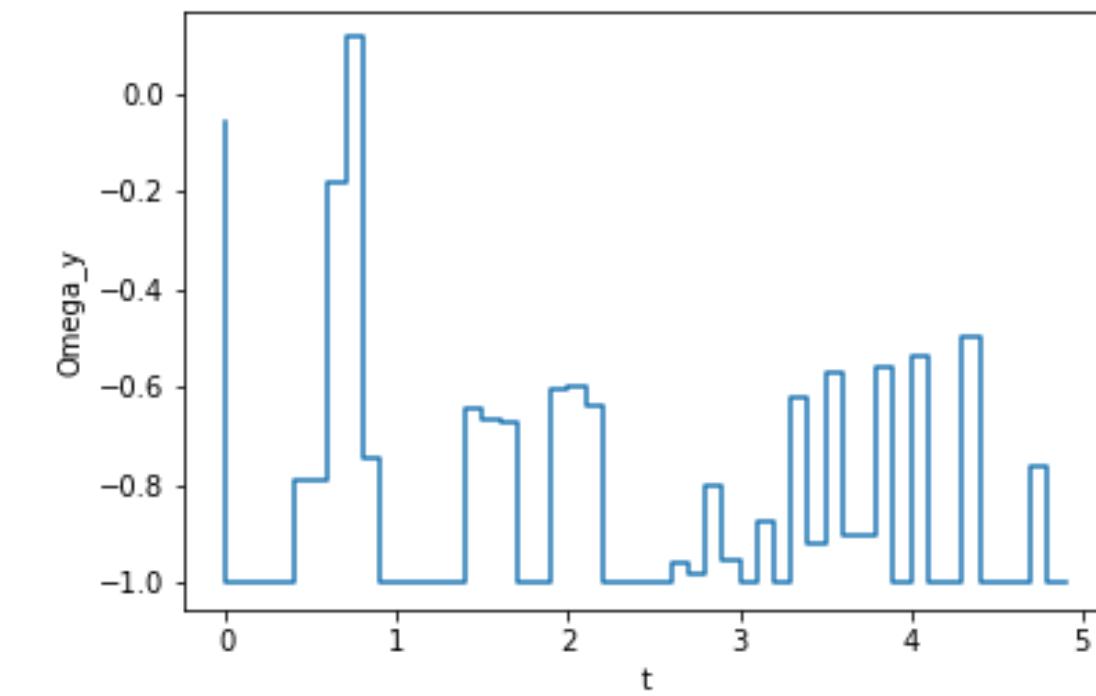
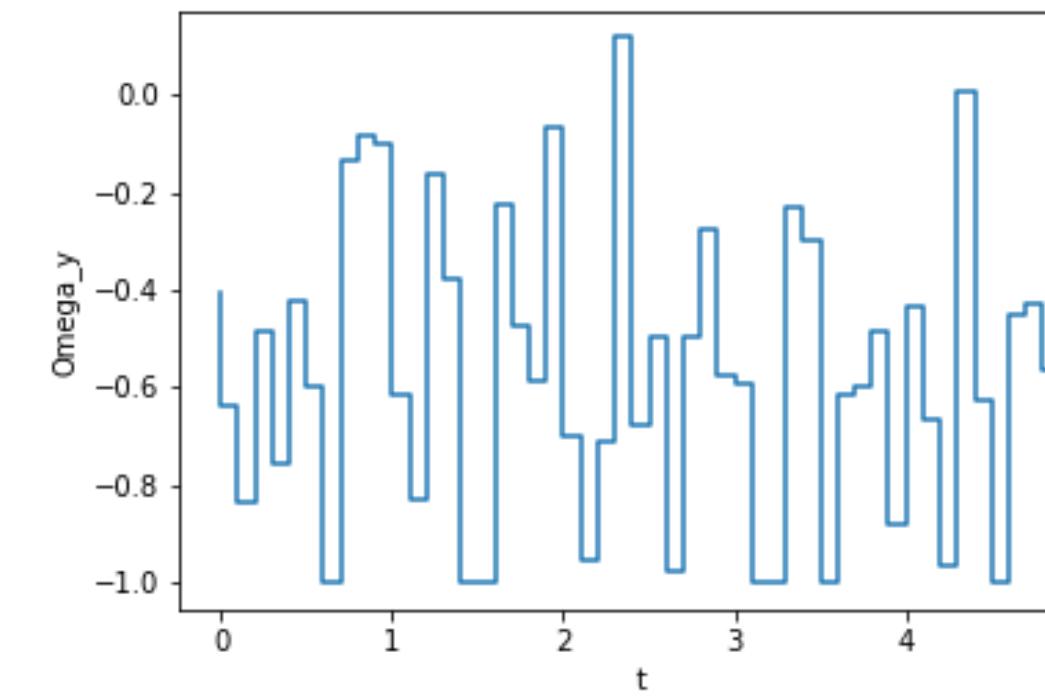
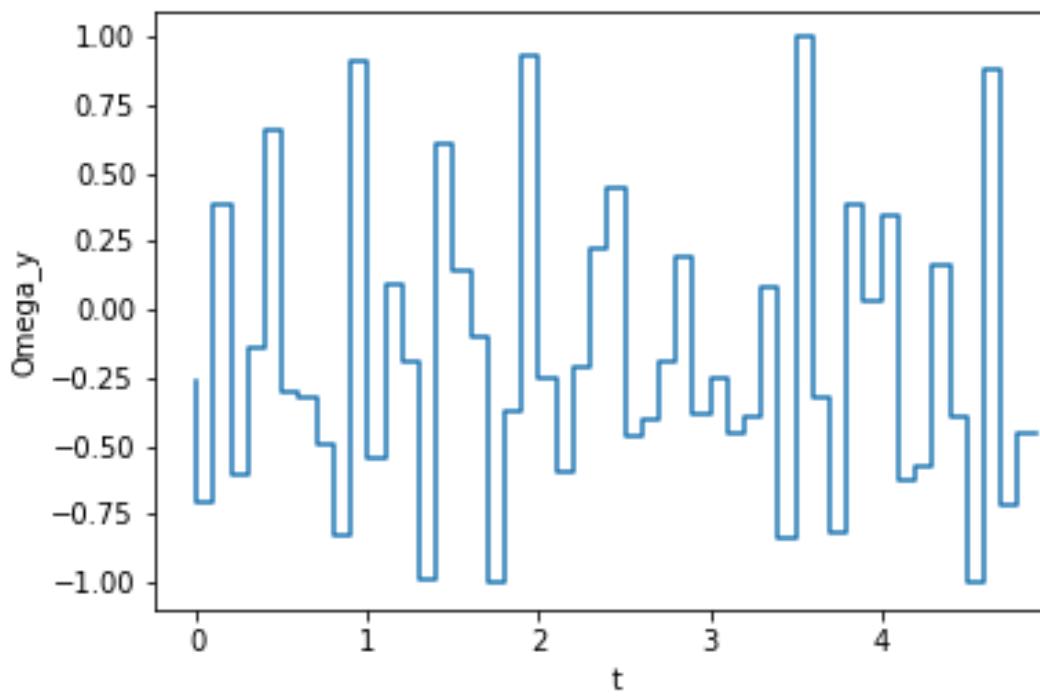
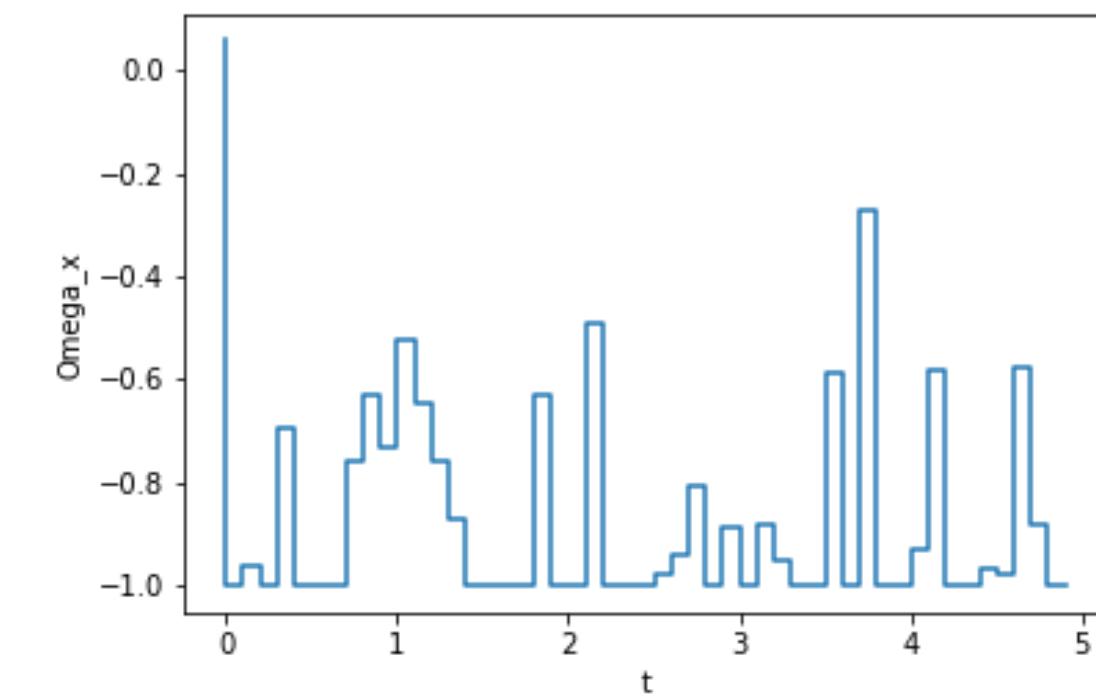
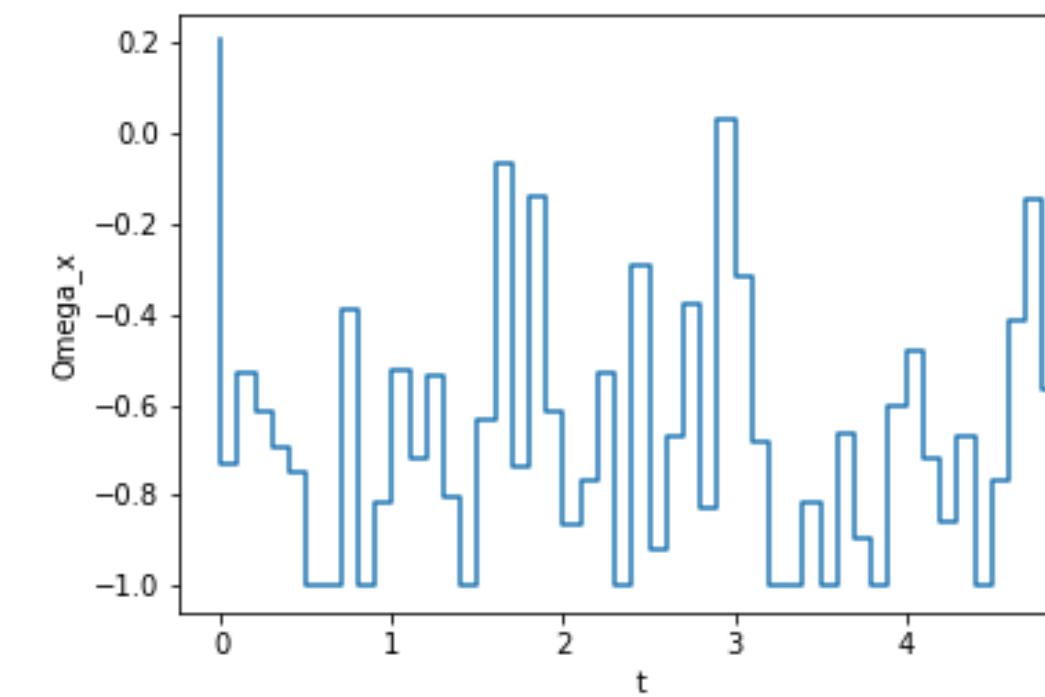
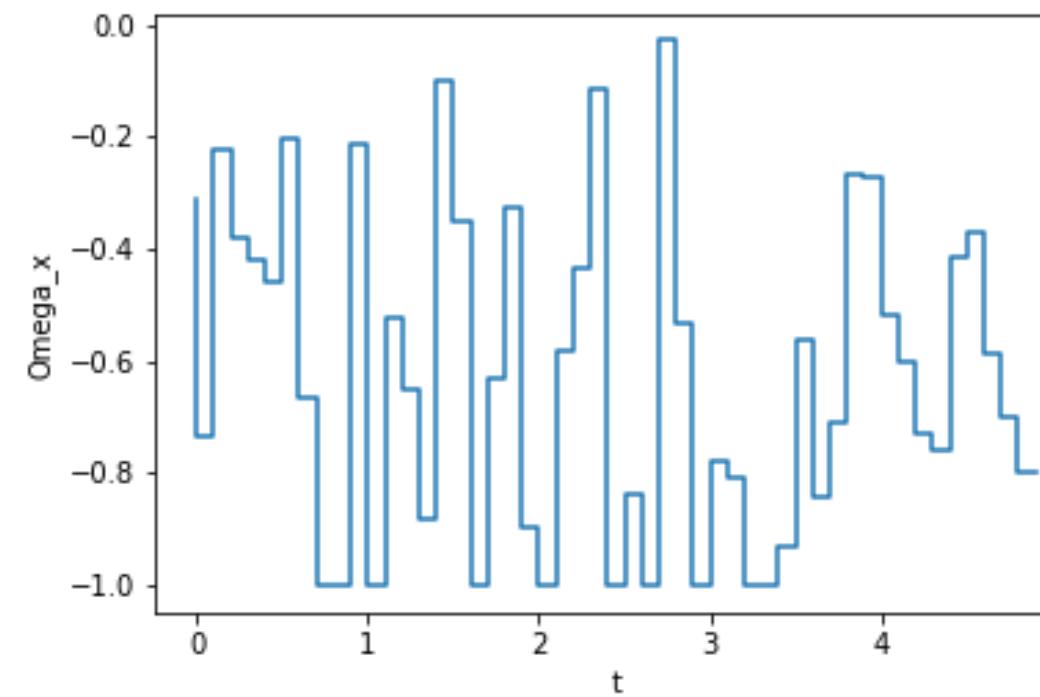
$$\gamma_1 = 0 \quad \gamma_2 = 0.1$$

$\Omega = 1$



$\frac{\Delta}{\Omega} = 0.1$

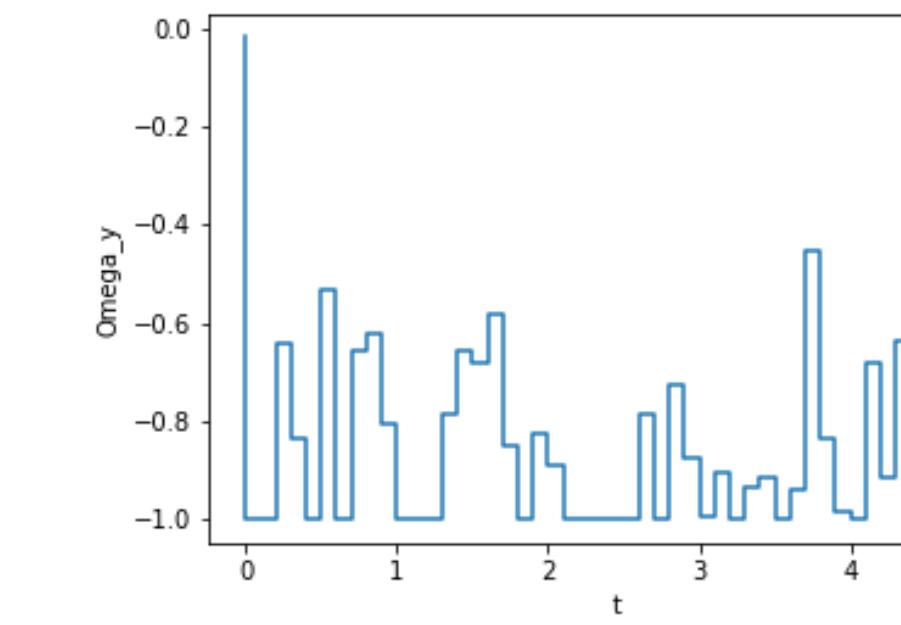
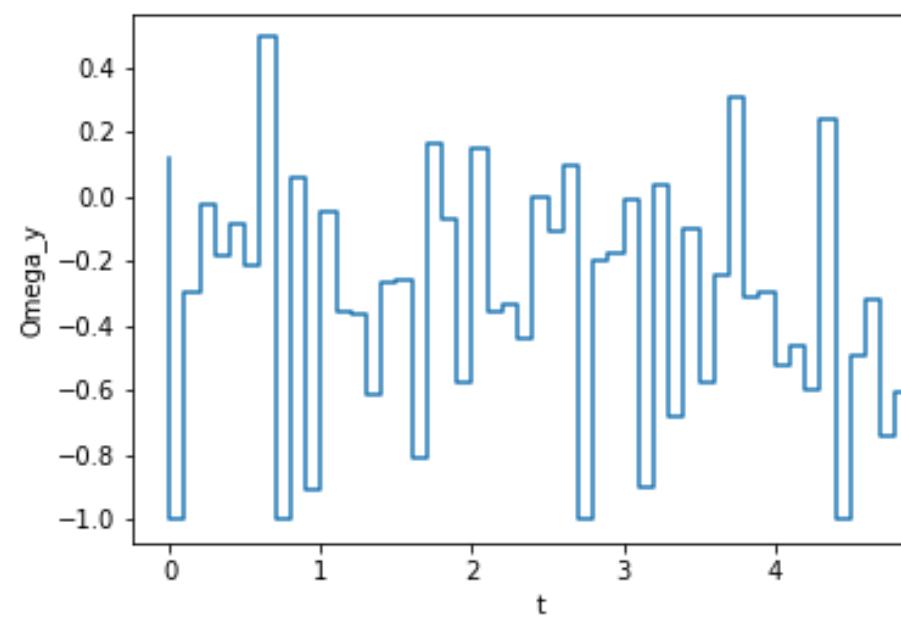
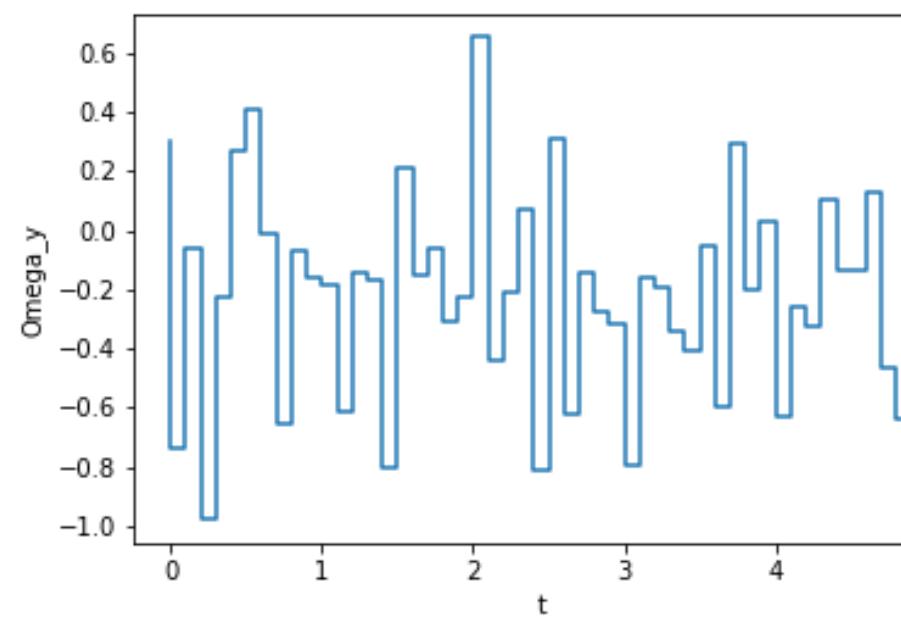
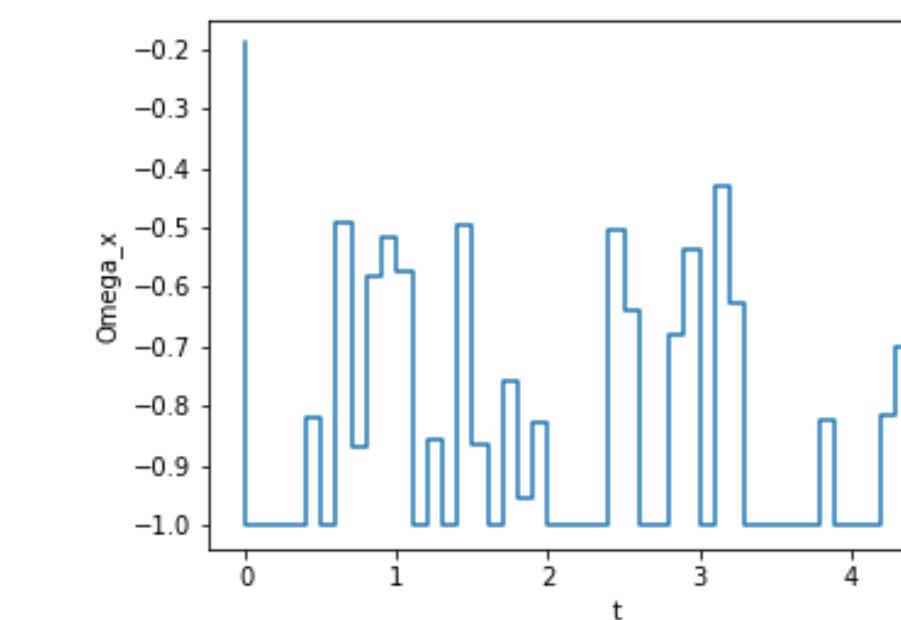
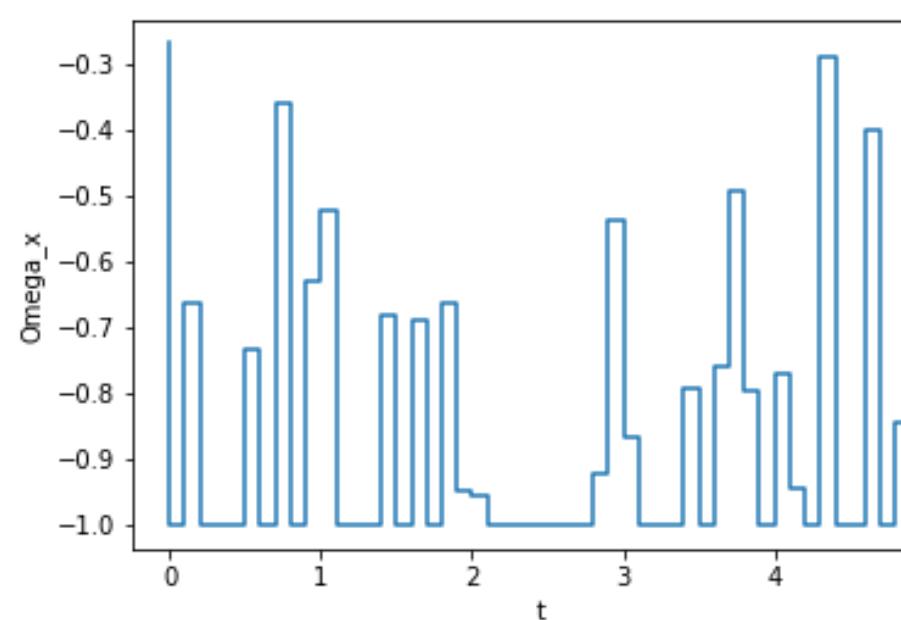
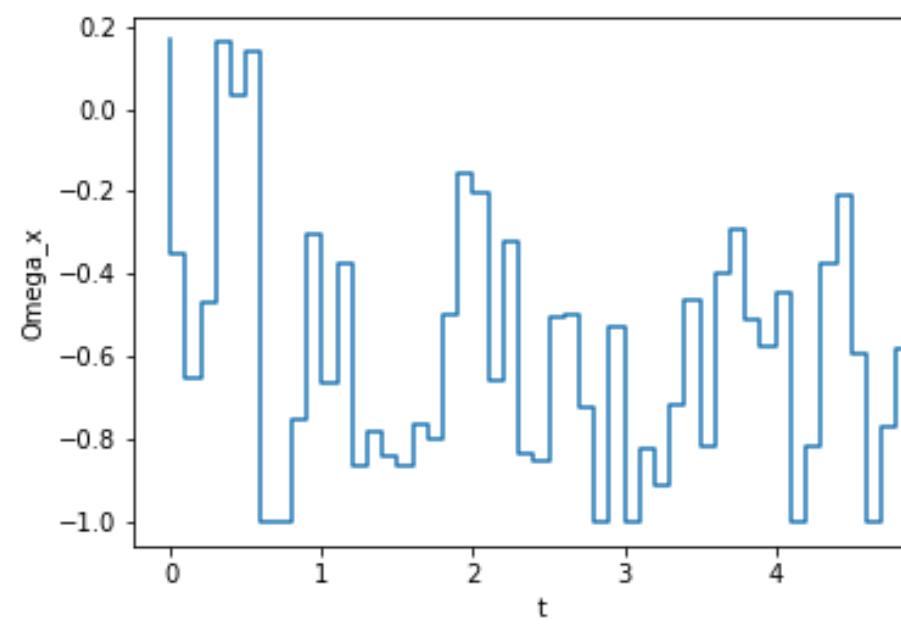
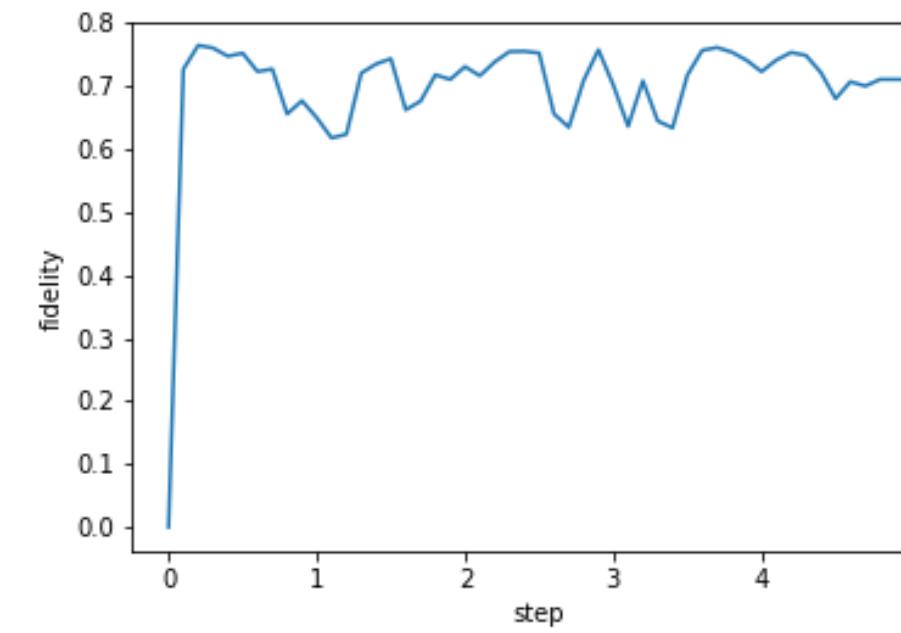
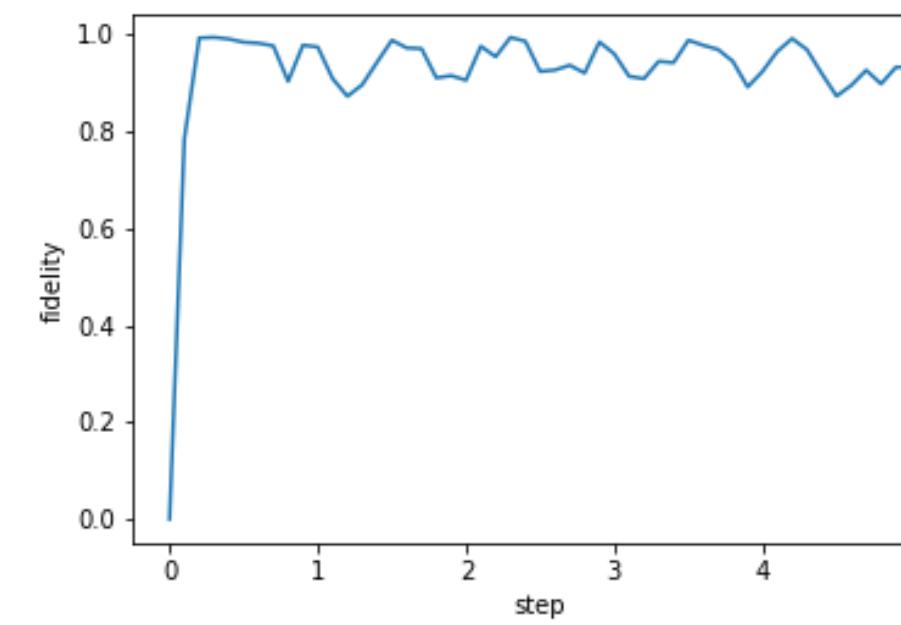
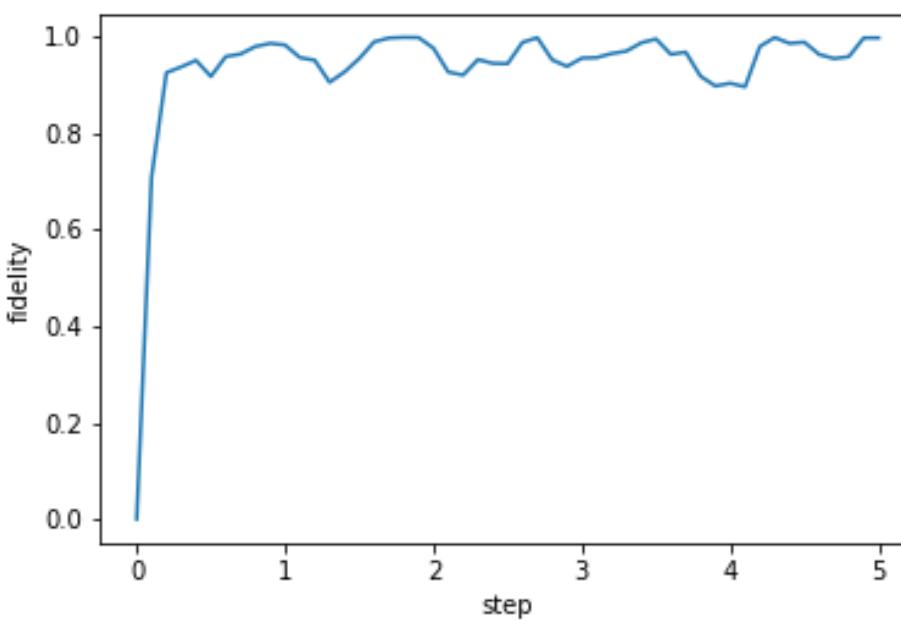
$T = 5 T_\pi$



$\gamma_1 = 0$

$\gamma_1 = 10$

$\gamma_1 = 100$

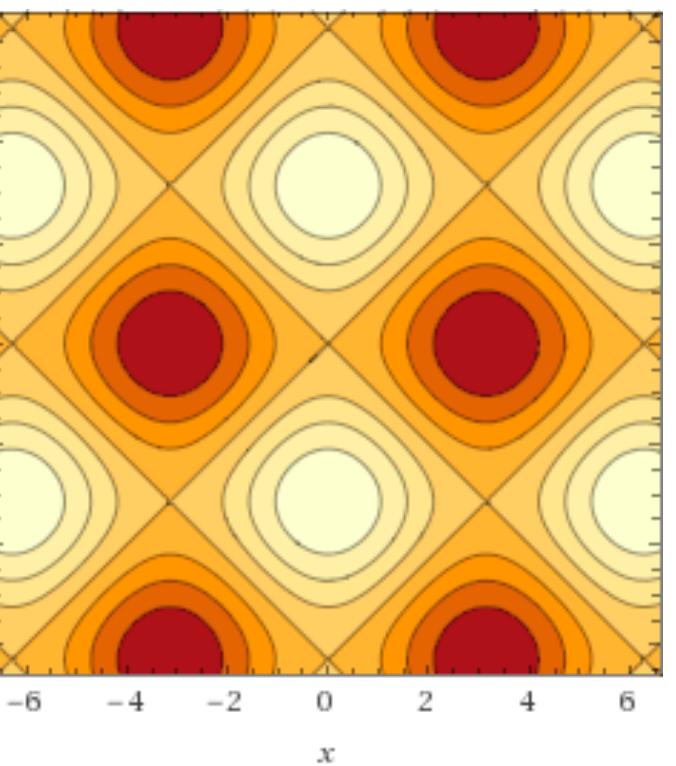
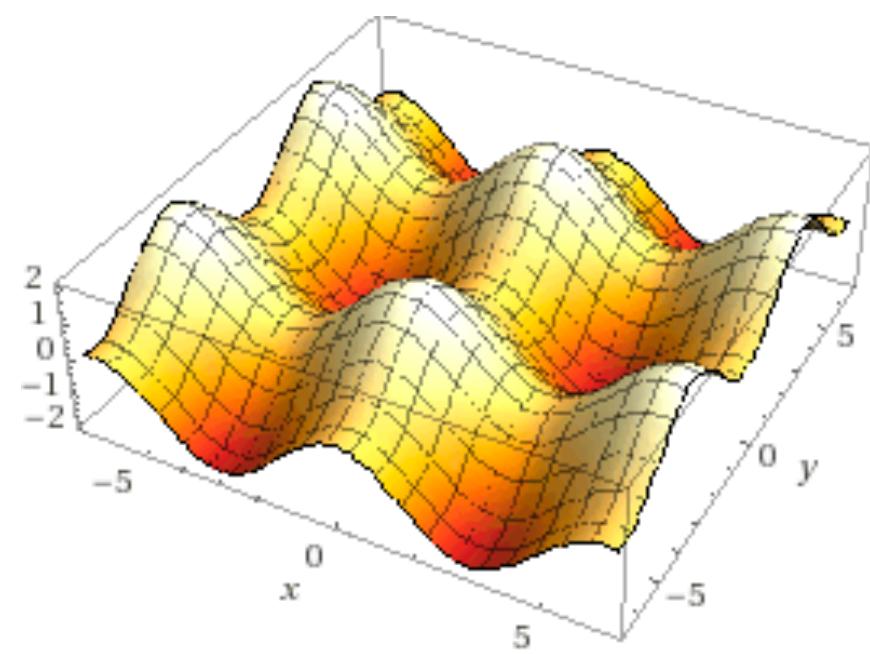
$\Omega = 1$ $\frac{\Delta}{\Omega} = 0.1$ $T = 5 T_\pi$  $\gamma_2 = 1$ $\gamma_2 = 10$ $\gamma_2 = 100$

An Open Question and Probable Interpretation

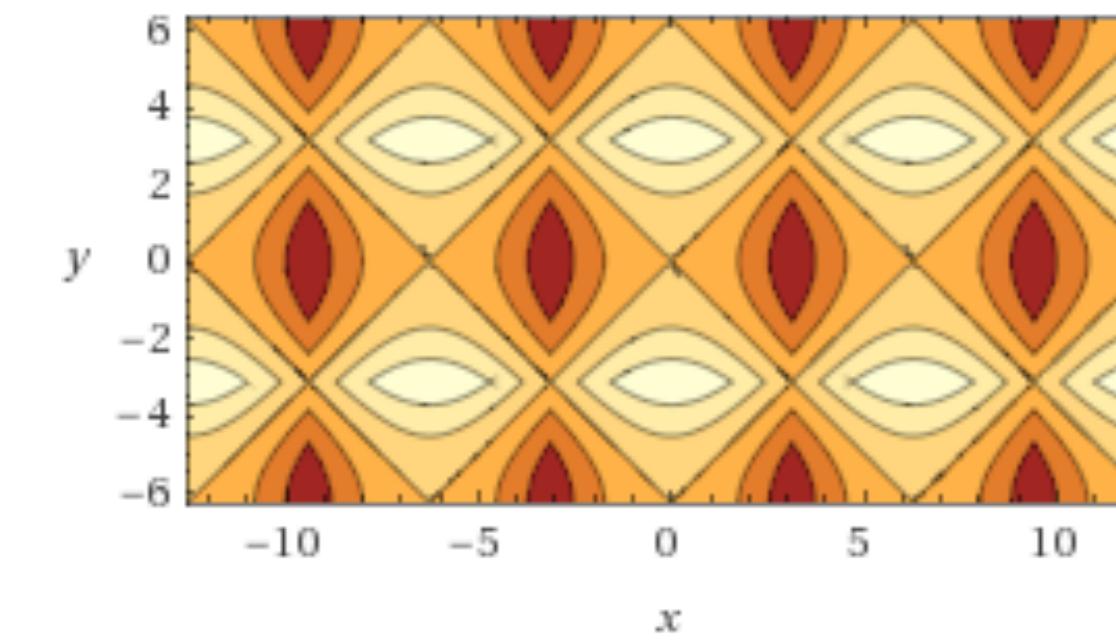
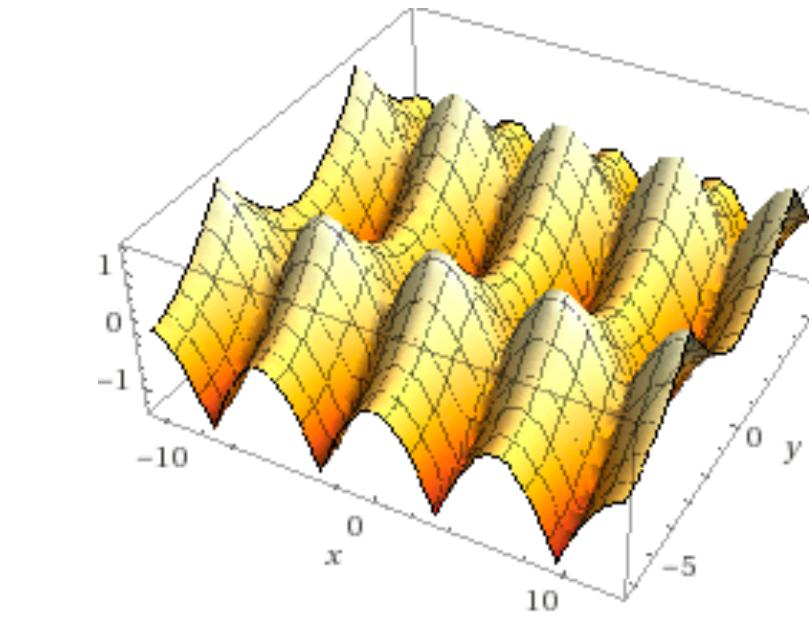
Fidelity

vs

$\sqrt{Fidelity}$



$$\cos(x) - \cos(y)$$



$$\sqrt{1 + \cos(x)} - \sqrt{1 + \cos(y)}$$

Scope of Improvements

Teaching the impact of error for generalisation with

<expected_state | resultant_state>

Expected state : state expected by the action in the absence of noise

Resultant_state : Actual state resulted by the action

Florian Frank,¹ Thomas Unden,¹ Jonathann Zoller,² Ressa S. Said,²
Tommaso

Calarco,² Simone Montangero,^{2, 3} Boris Naydenov,¹ and Fedor
Jelezko¹

Silicon Quantum ElectronicsSusan N. Coppersmith, Mark A. Eriksson

Griffiths introduction to quantum mechanics

Resonantly driven CNOT gate for electron spins

D. M. Zajac,¹ A. J. Sigillito,¹ M. Russ,² F. Borjans,¹ J. M. Taylor,^{3,4}

Reinforcement Learning and Optimal Control

Dimitri P. Bertsekas Massachusetts Institute of Technology

A Survey of Optimization Methods from