Can a machine infer quantum dynamics?

Training a Recurrent Neural Network to Predict Quantum Trajectories from Raw Observation.

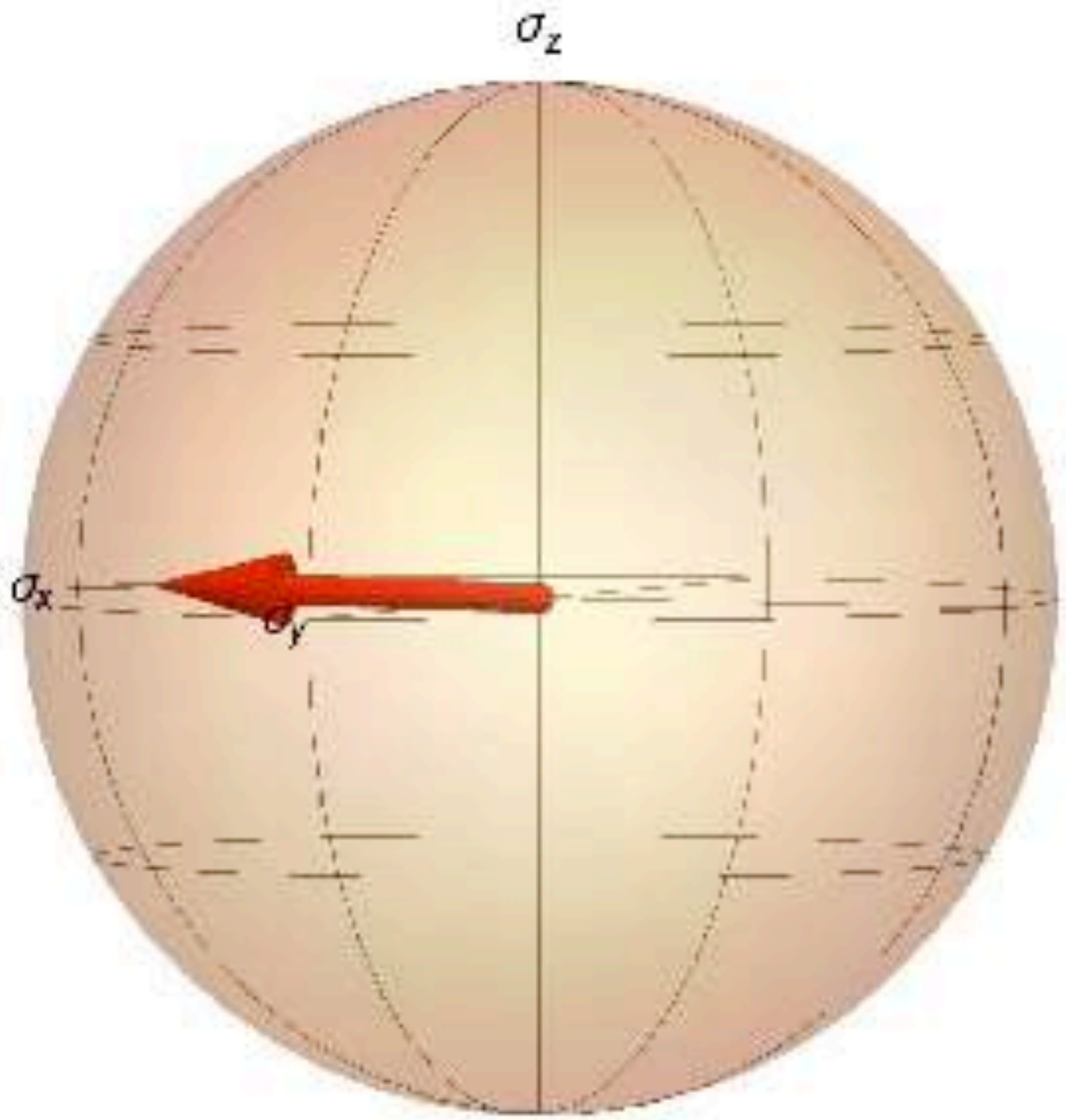
arXiv:1811.12420

Emmanuel Flurin*

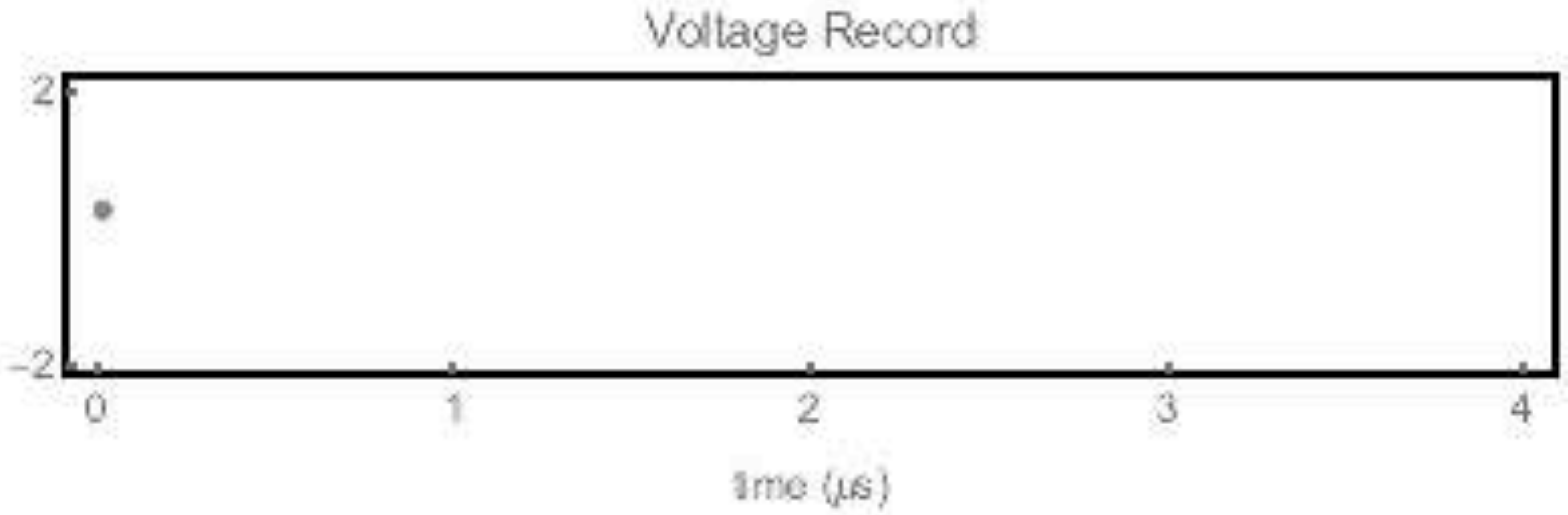
S. Hacohen-Gourgy, L. Martin, I. Siddiqi

Quantum Nanoelectronics Laboratory, UC Berkeley

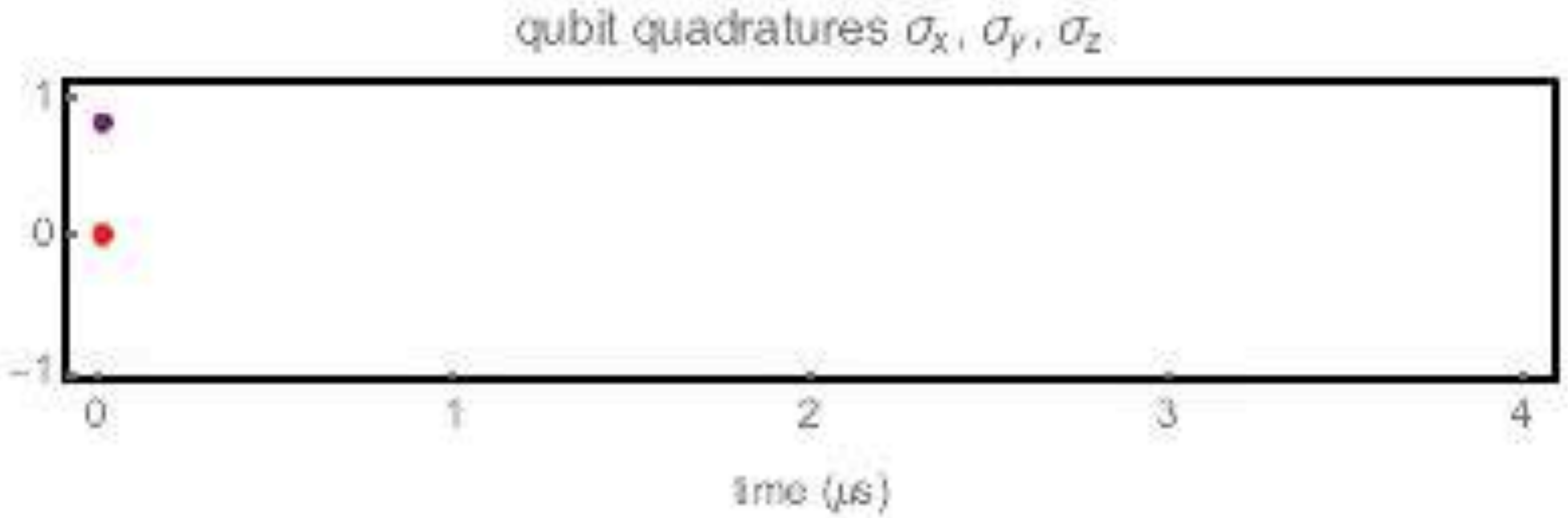
*Quantronics, CEA Saclay, France



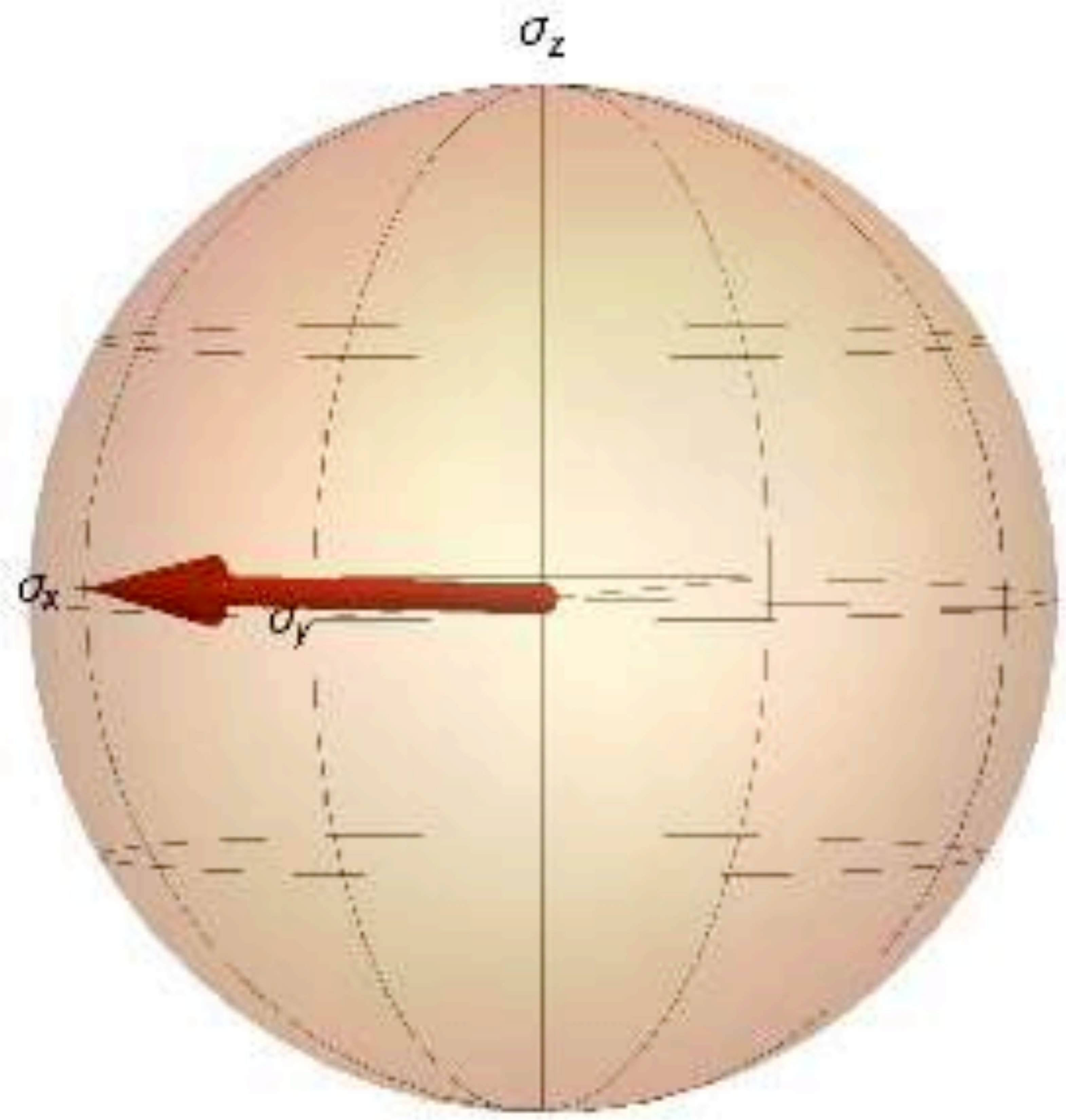
σ_z
σ_x
σ_y



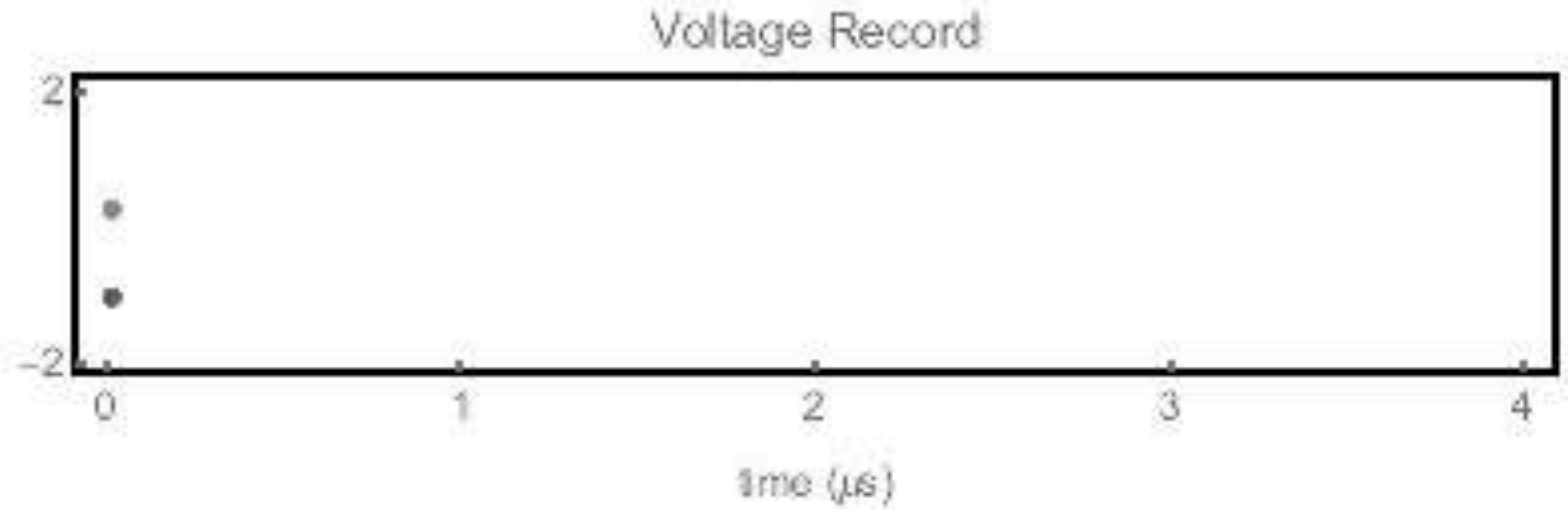
Voltage Record
2
−2
0
1
2
3
4
time (μs)



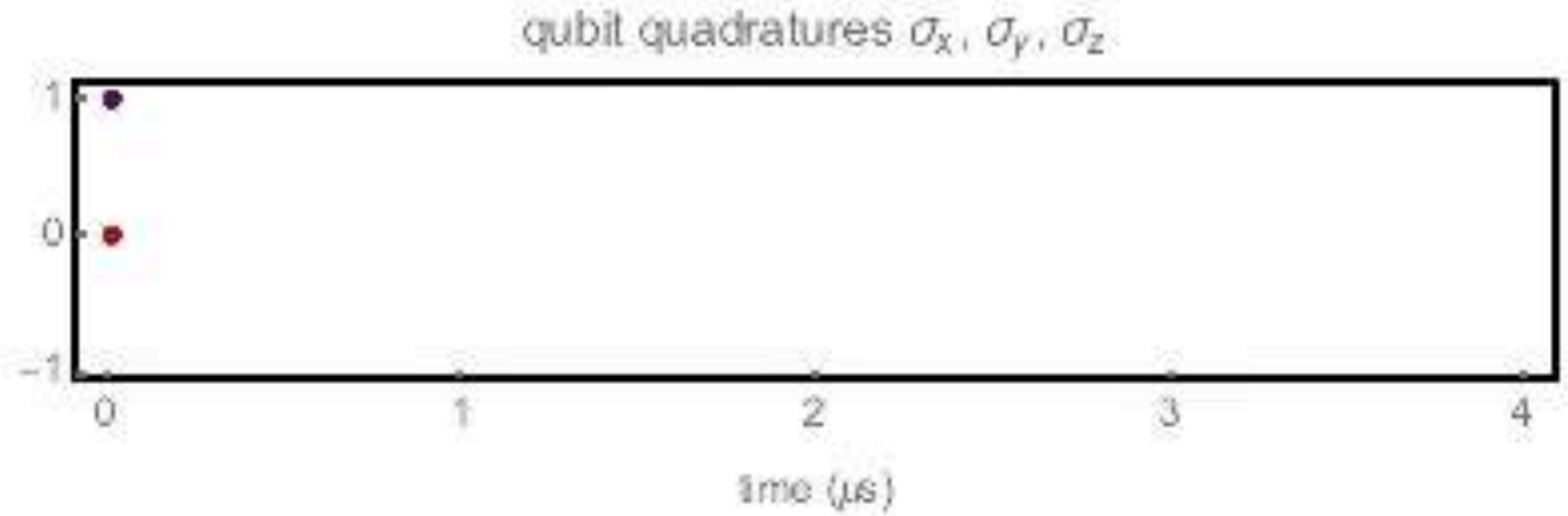
qubit quadratures σ_x, σ_y, σ_z
1
0
−1
0
1
2
3
4
time (μs)



σ_z
σ_x
σ_y

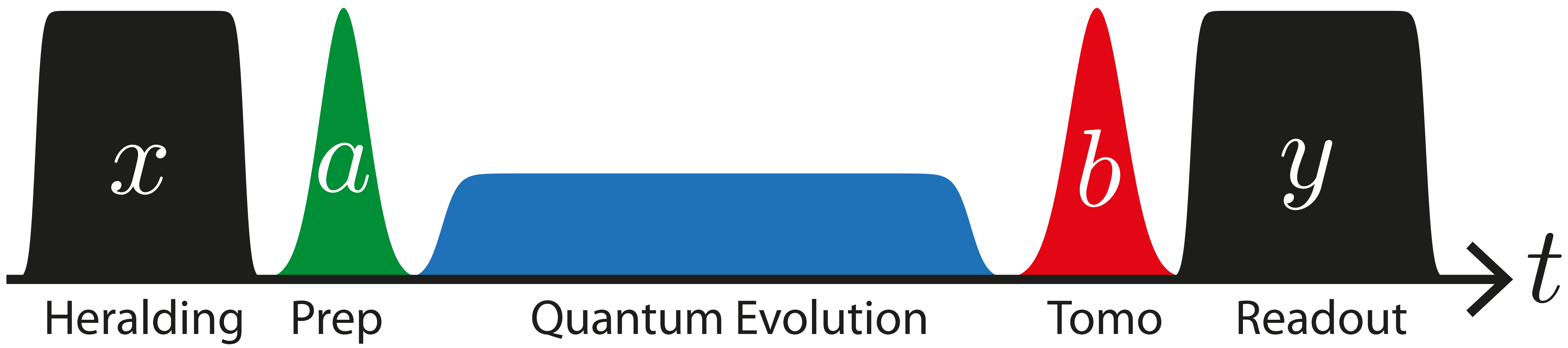


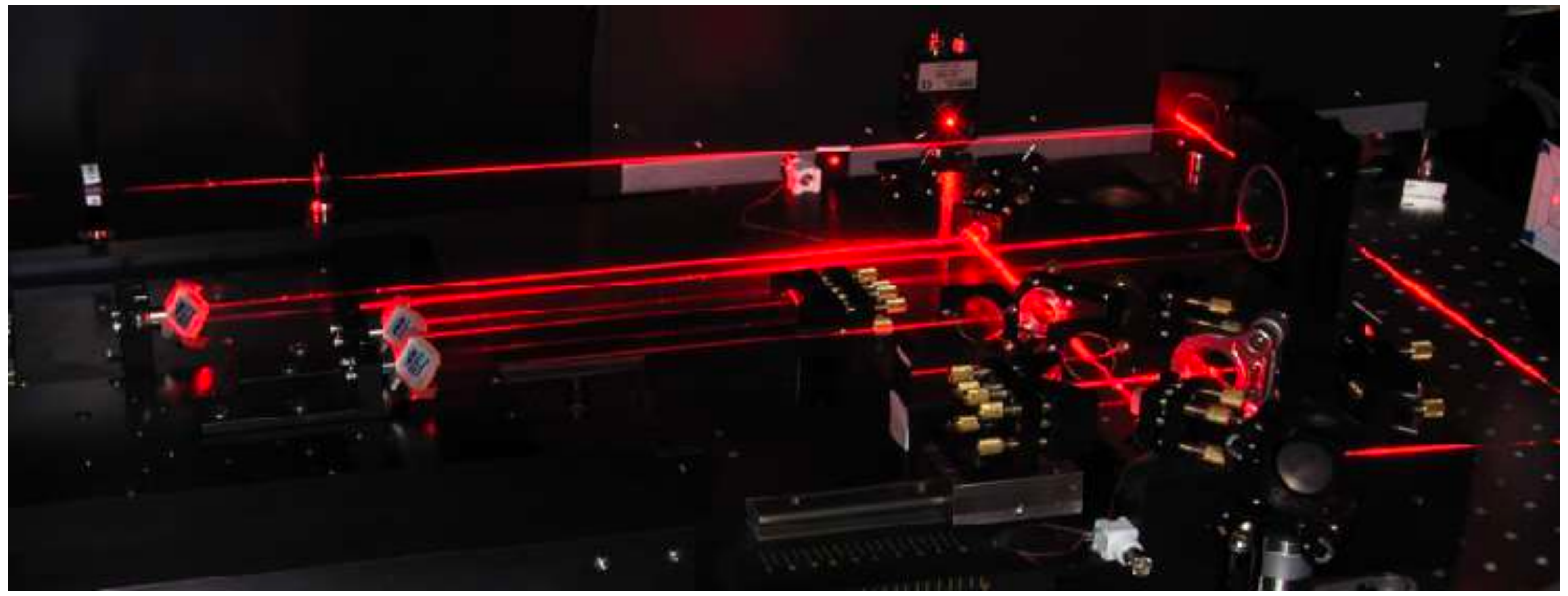
Voltage Record
2
−2
0
1
2
3
4
time (μs)

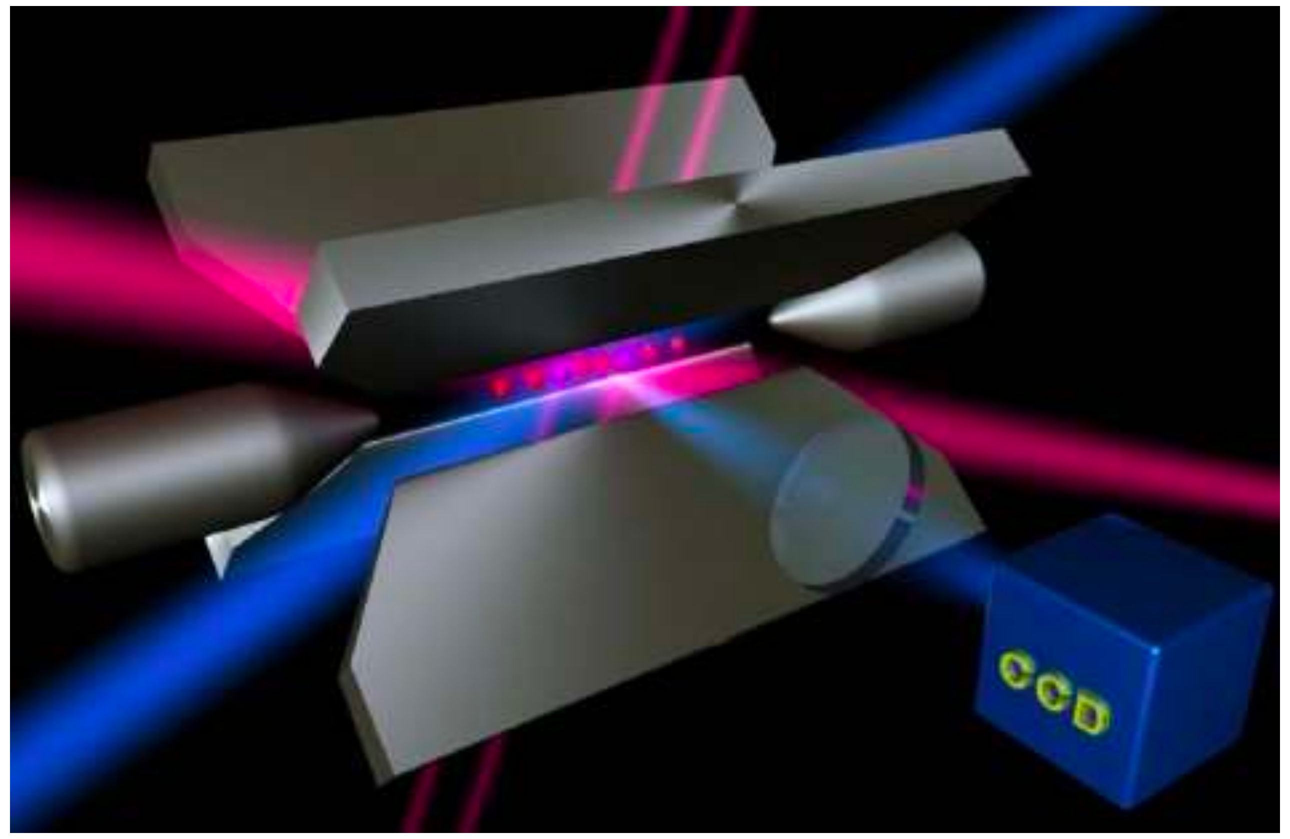


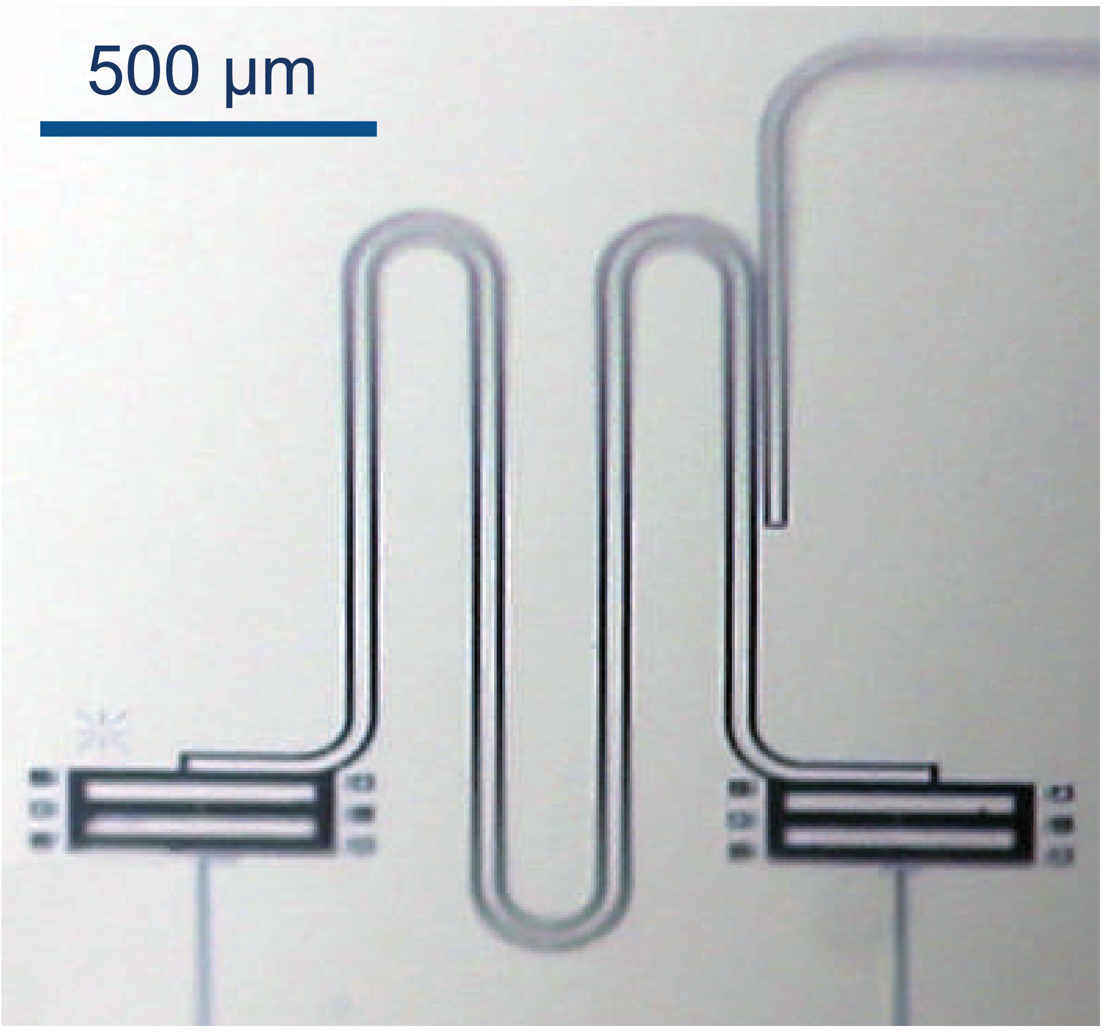
qubit quadratures σ_x, σ_y, σ_z
1
0
−1
0
1
2
3
4
time (μs)

A Quantum Information Processing Experiment

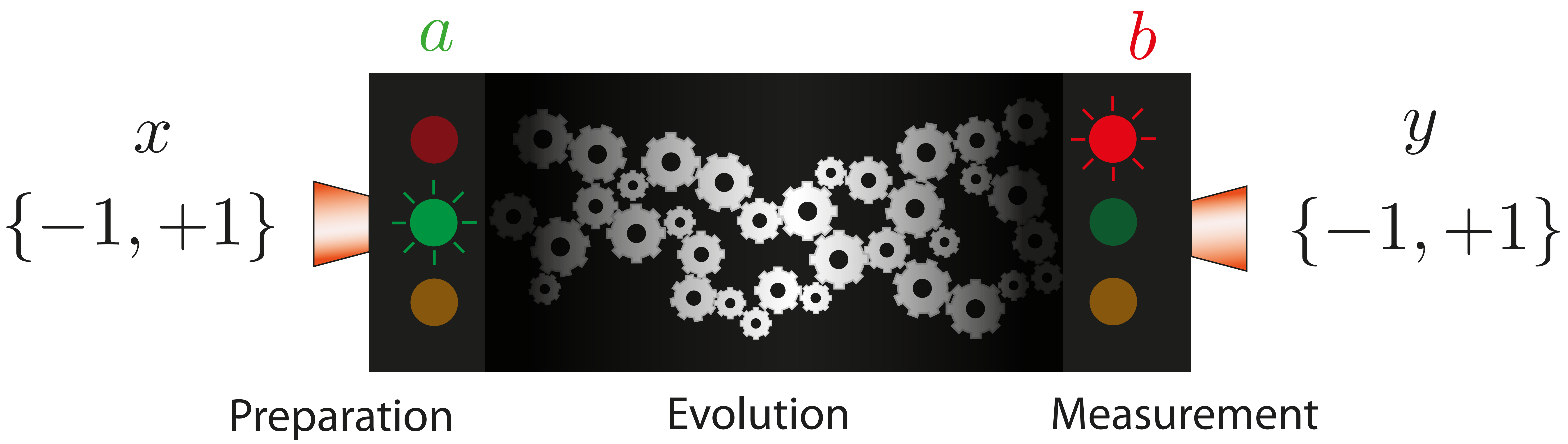




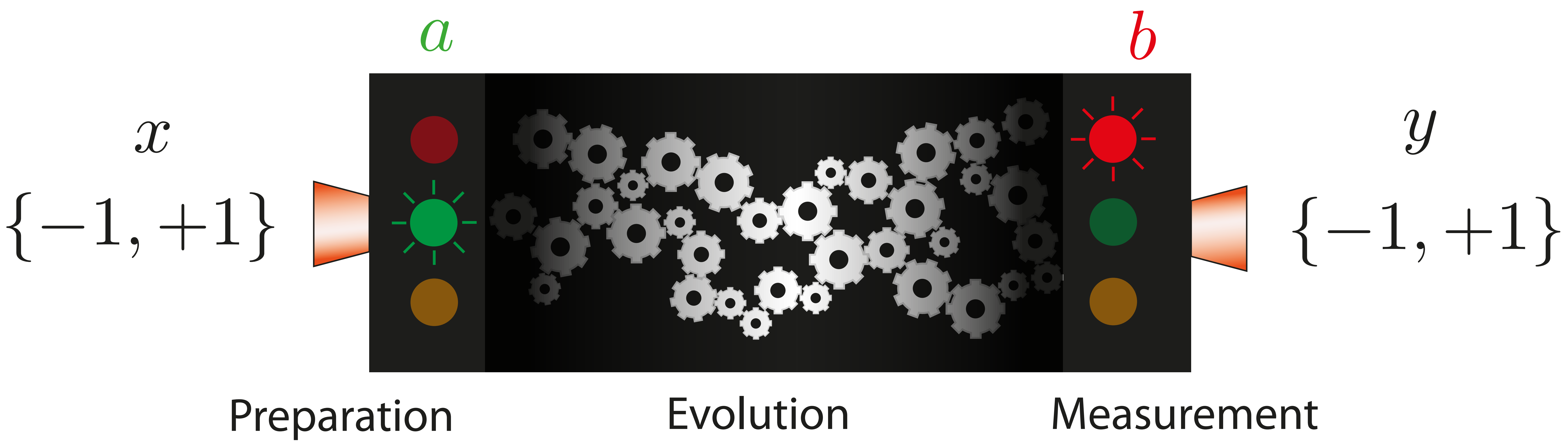




Black Box Quantum Mechanics



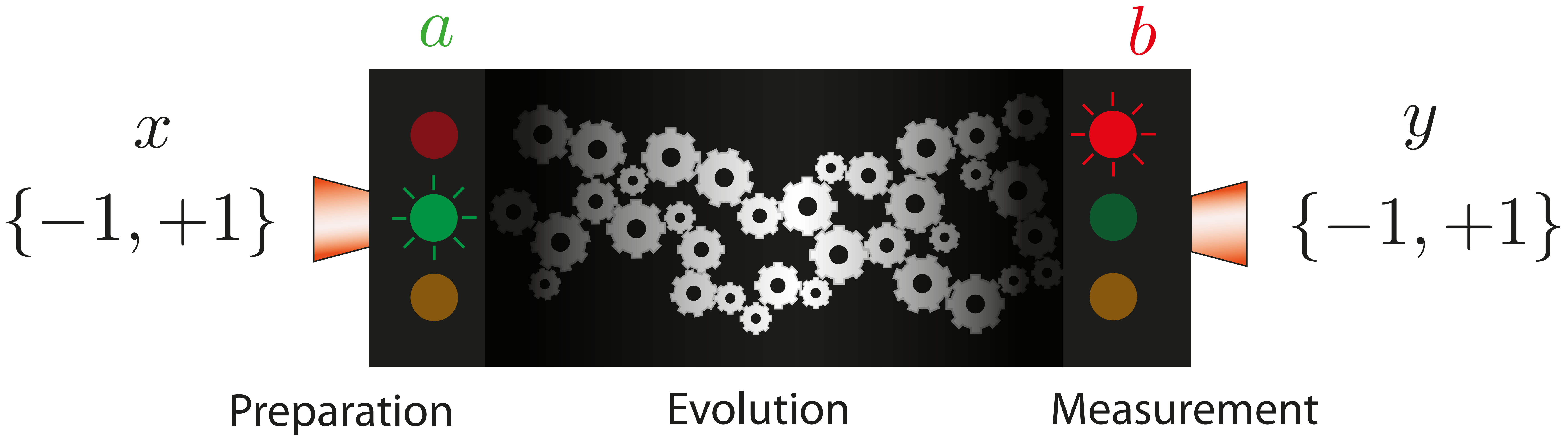
Black Box Quantum Mechanics



Quantum mechanics gives

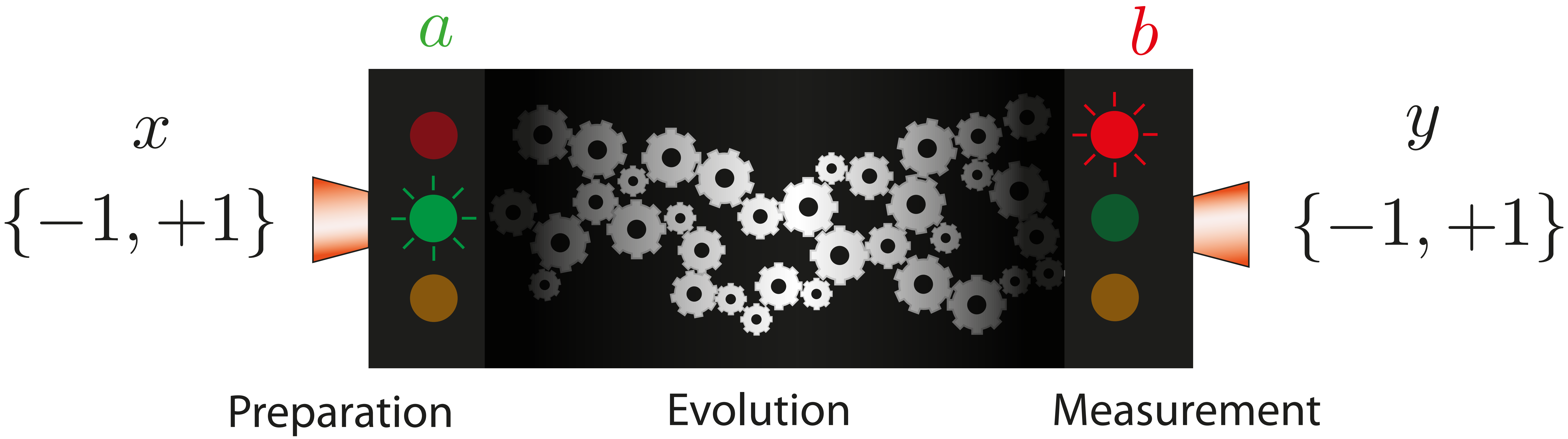
$$P(y|x, a, b)$$

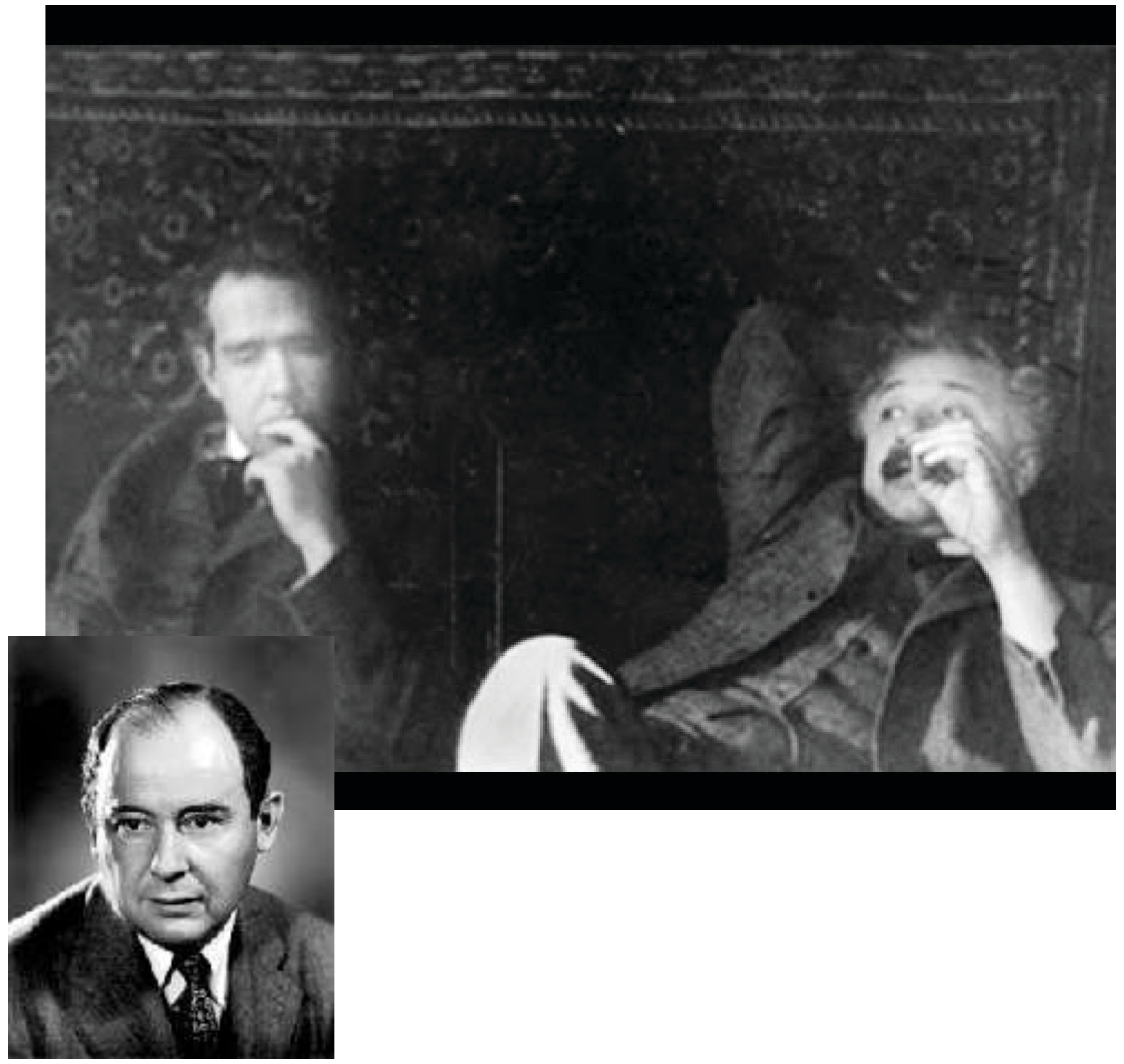
Black Box Quantum Mechanics



$$P(y|x,{\color{green}a},{\color{red}b}) = |\langle y|{\color{red}\hat{B}}e^{-i\hat{H}t}{\color{green}\hat{A}}|x\rangle|^2$$

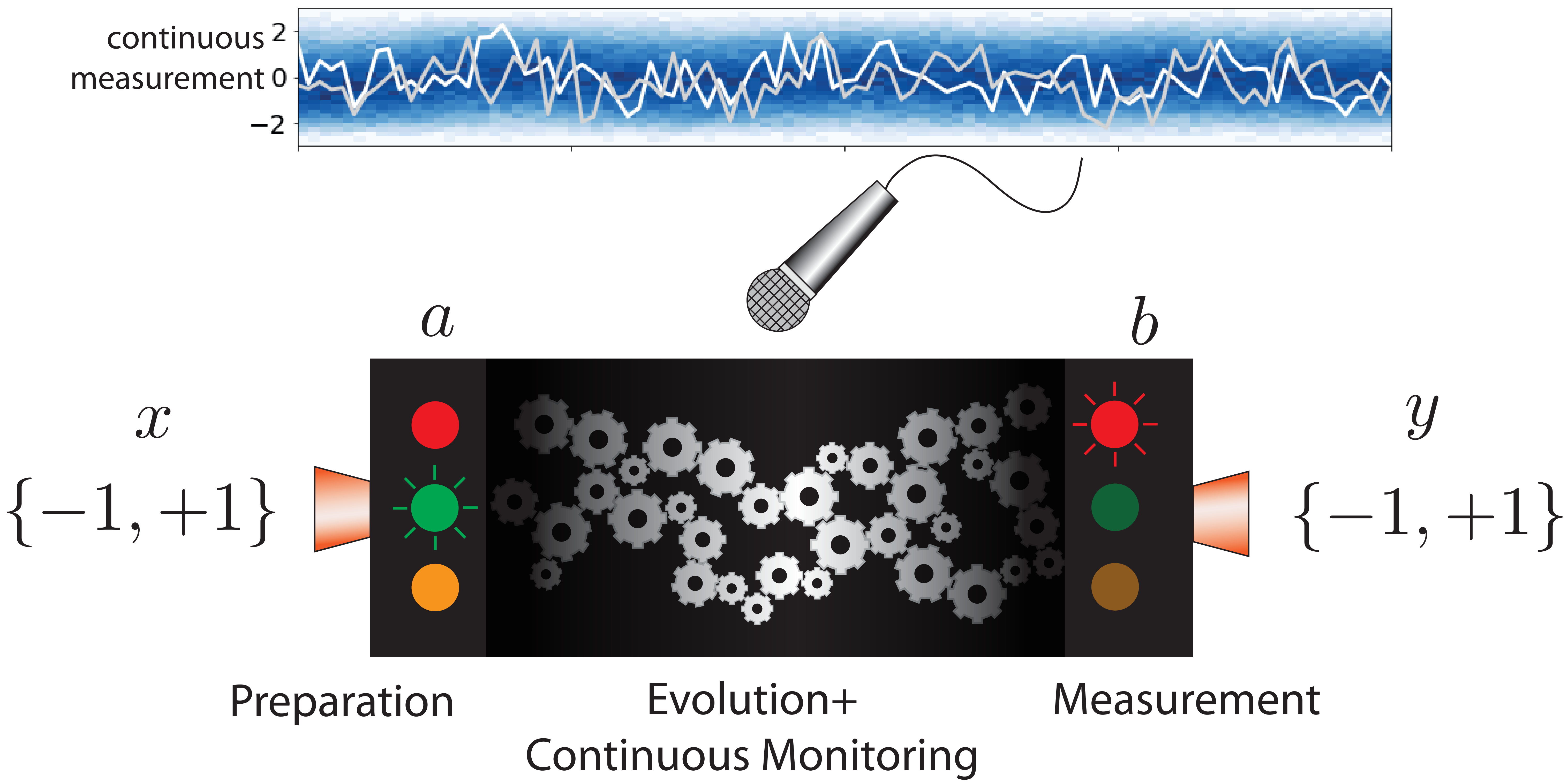
Black Box Quantum Mechanics





$$P(y|x,a,b) = \mathrm{Tr}(|y\rangle\langle y|\hat{B}e^{-i\hat{H}t}\hat{A}\rho_x\hat{A}^\dagger e^{i\hat{H}t}\hat{B}^\dagger)$$

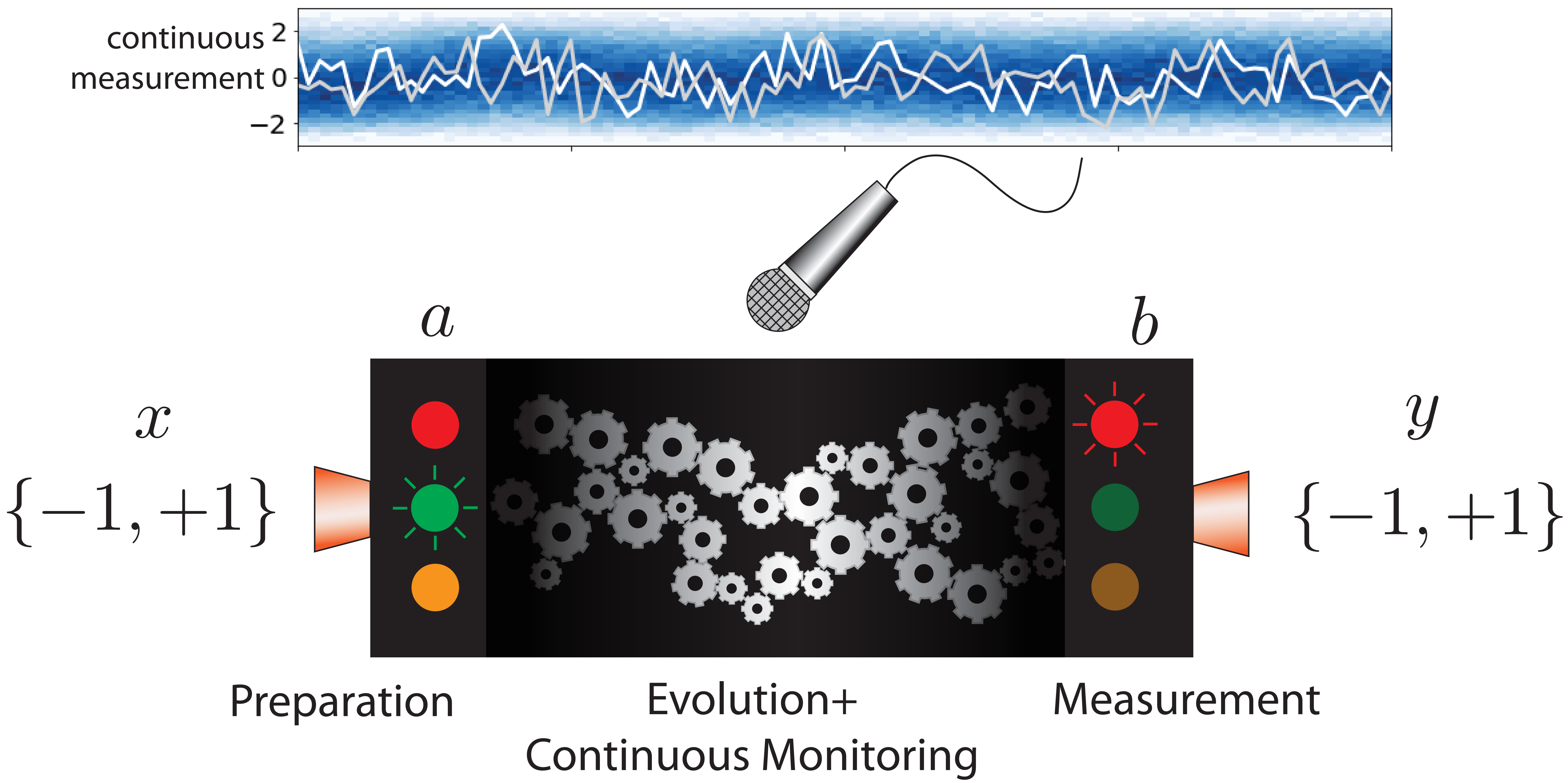
Stochastic Quantum Evolution



Quantum mechanics gives

$$P(y|x, a, b, V_0, \ldots, V_t, \ldots, V_T) =$$

Stochastic Quantum Evolution



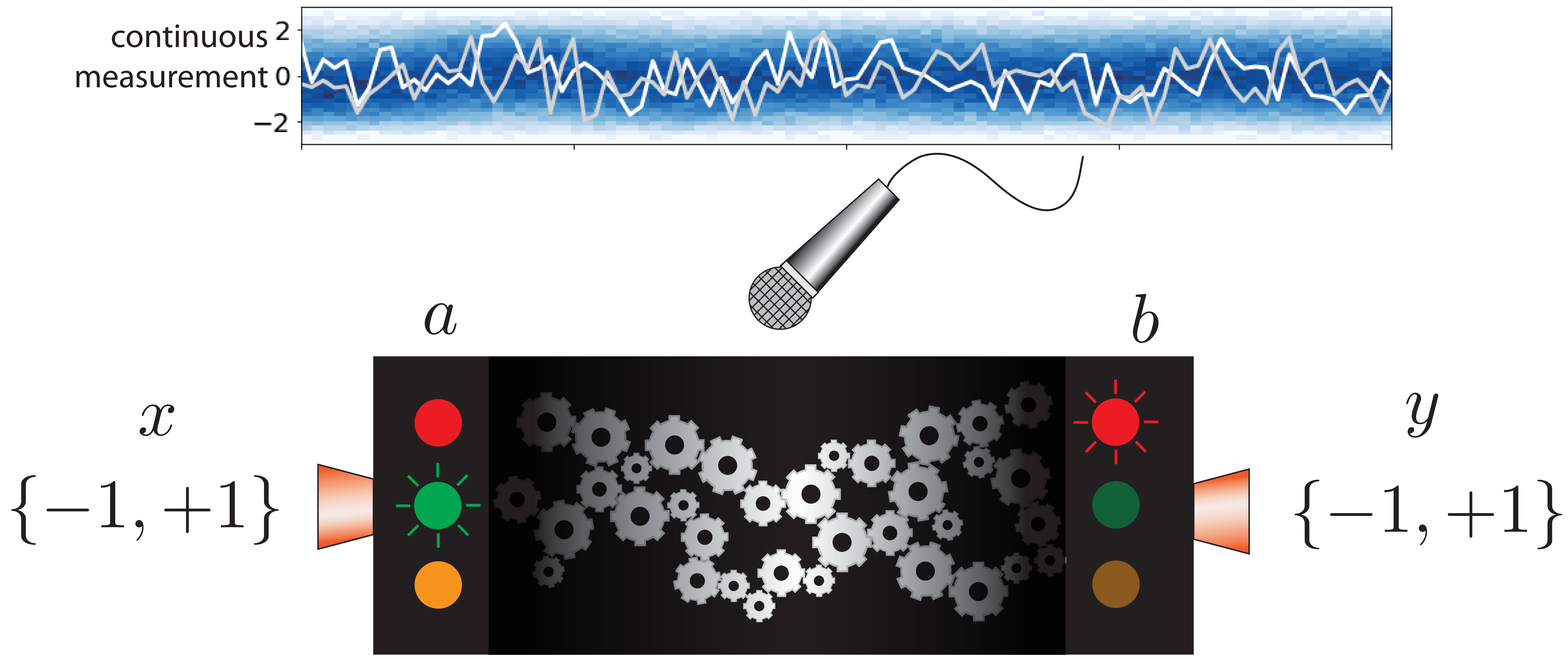
« POVM »
measurement backaction +
coherent dynamics

$$P(y|x, a, b, V_0, ..., V_t, ..., V_T) =$$

$$\frac{\mathrm{Tr}(|y\rangle\langle y|\hat{B}\hat{\Omega}_{V_T}\ldots\hat{\Omega}_{V_t}\ldots\hat{\Omega}_{V_0}\hat{A}\rho_x\hat{A}^\dagger\hat{\Omega}^\dagger_{V_0}\ldots\hat{\Omega}^\dagger_{V_t}\ldots\hat{\Omega}^\dagger_{V_T}\hat{B}^\dagger)}{\mathrm{Tr}(\hat{\Omega}_{V_T}\ldots\hat{\Omega}_{V_t}\ldots\hat{\Omega}_{V_0}\hat{A}\rho_x\hat{A}^\dagger\hat{\Omega}^\dagger_{V_0}\ldots\hat{\Omega}^\dagger_{V_t}\ldots\hat{\Omega}^\dagger_{V_T})}$$

Physical parameters have to be separately calibrated and fine-tuned

Inferring Quantum Dynamics



If one have a large set of instances $(y, x, a, b, V_0, \ldots V_T)$

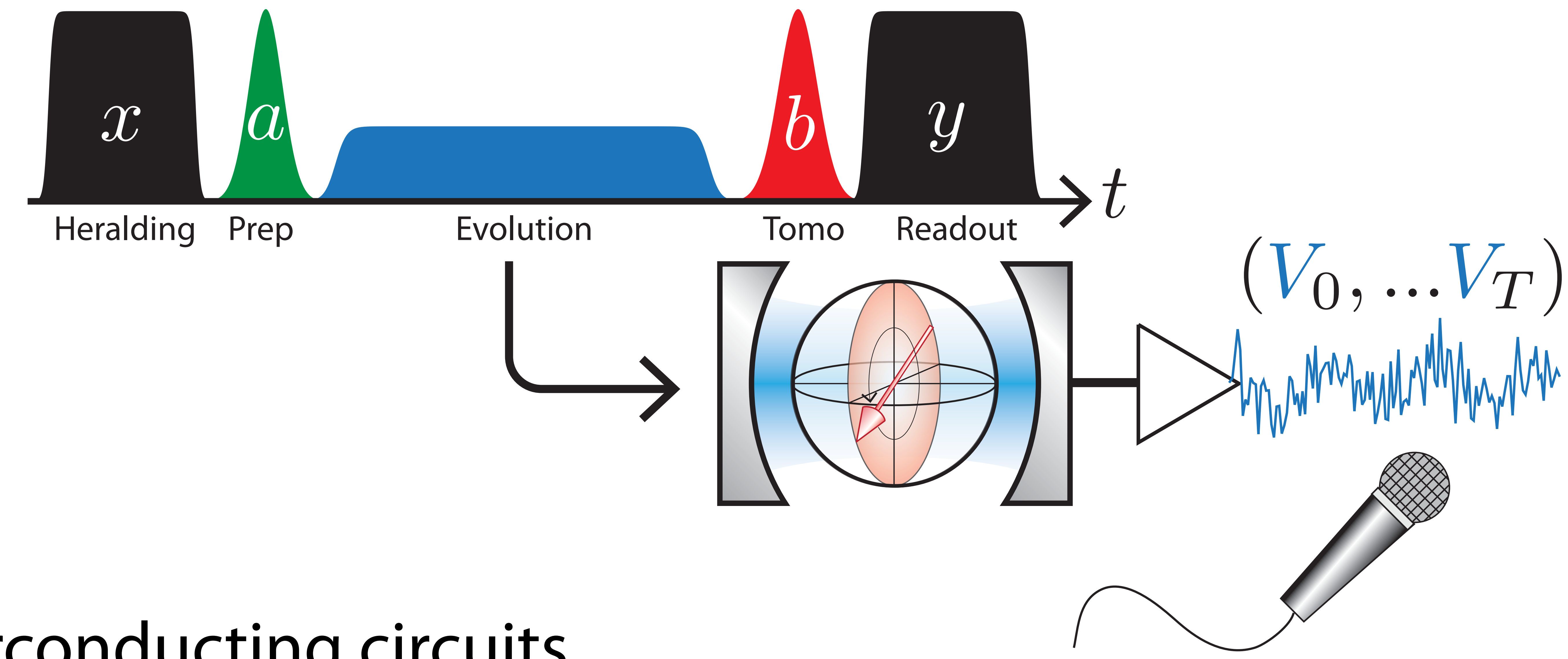
Supervised Deep Learning learns

$$P(y|x, a, b, V_0, \ldots, V_t, \ldots, V_T)$$

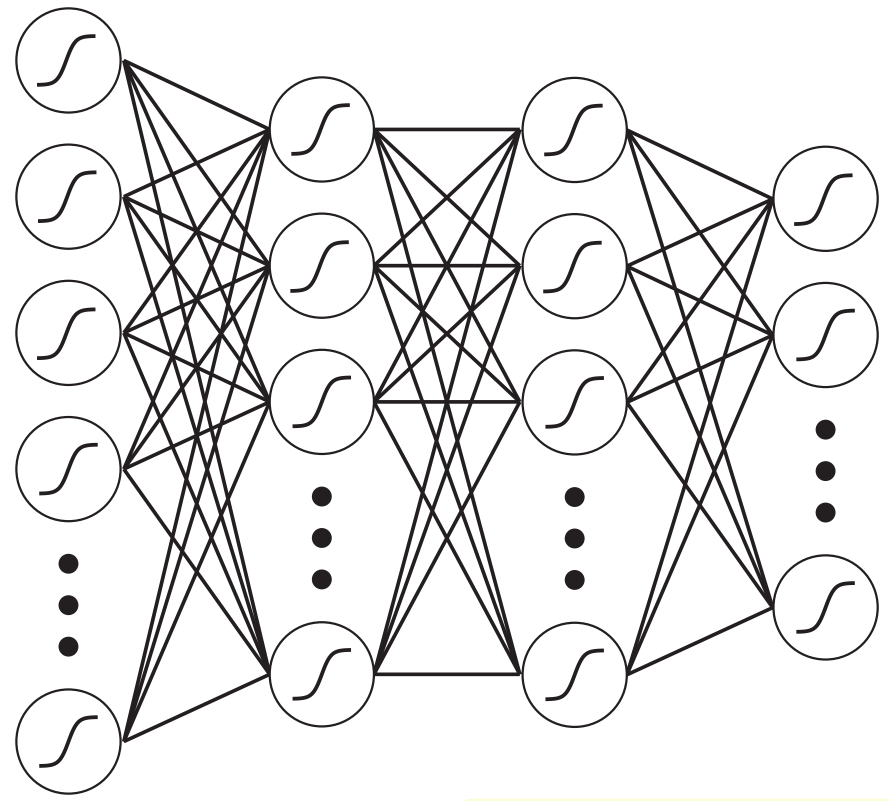
no matter how complicated the problem is

- language translation
- image & speech recognition
- medical diagnosis
- LHC signal processing

Inferring Quantum Dynamics



Superconducting circuits provides 10^6 instances $(y, x, a, b, V_0, \ldots V_T)$ per minutes



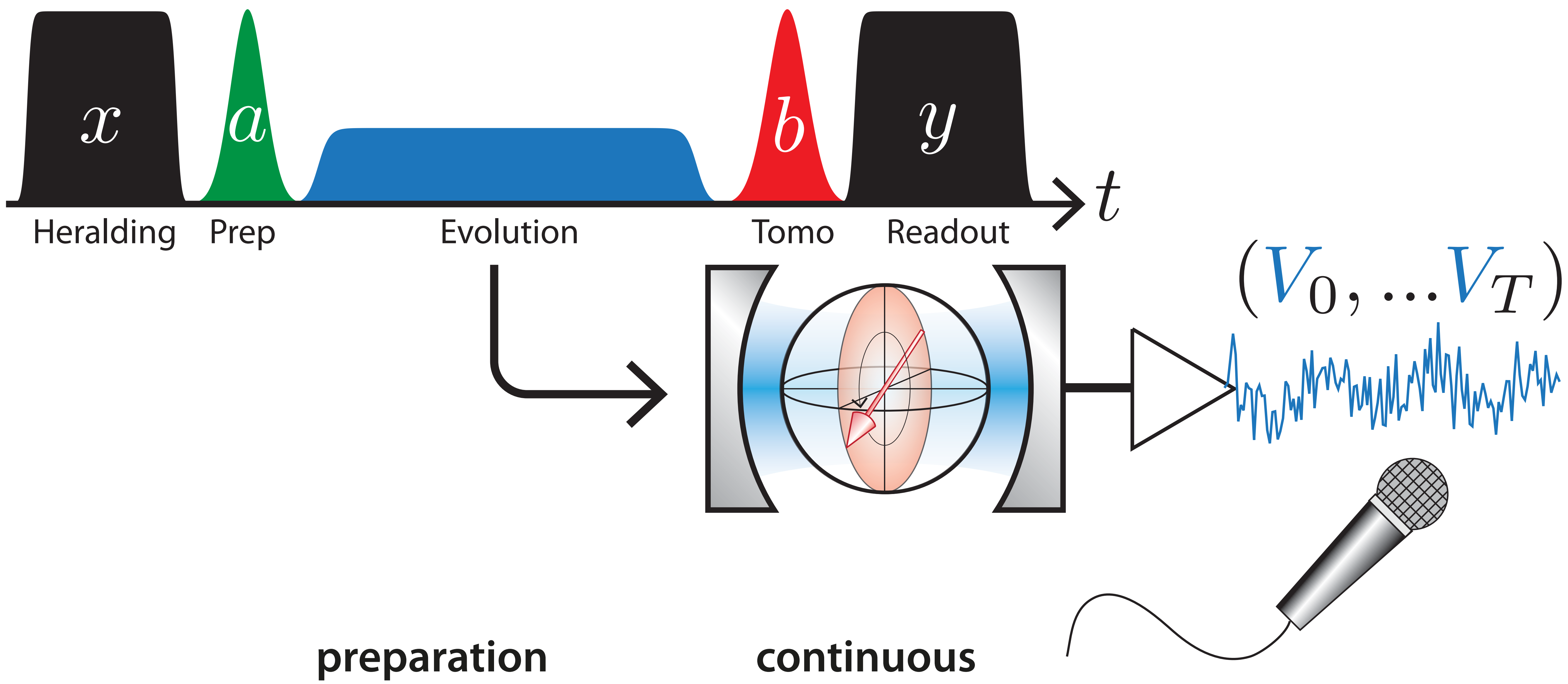
Deep neural network can learn

$$P(y|x, a, b, V_0, \ldots, V_t, \ldots, V_T)$$

with no prior on quantum mechanics

if b spans a complete set of observable, P is equivalent to the wave-function $\rho(T)$

Inferring Quantum Dynamics



preparation settings

continuous measurement record

data $\left(x, a, V_0, \ldots, V_t, \ldots V_T\right)$

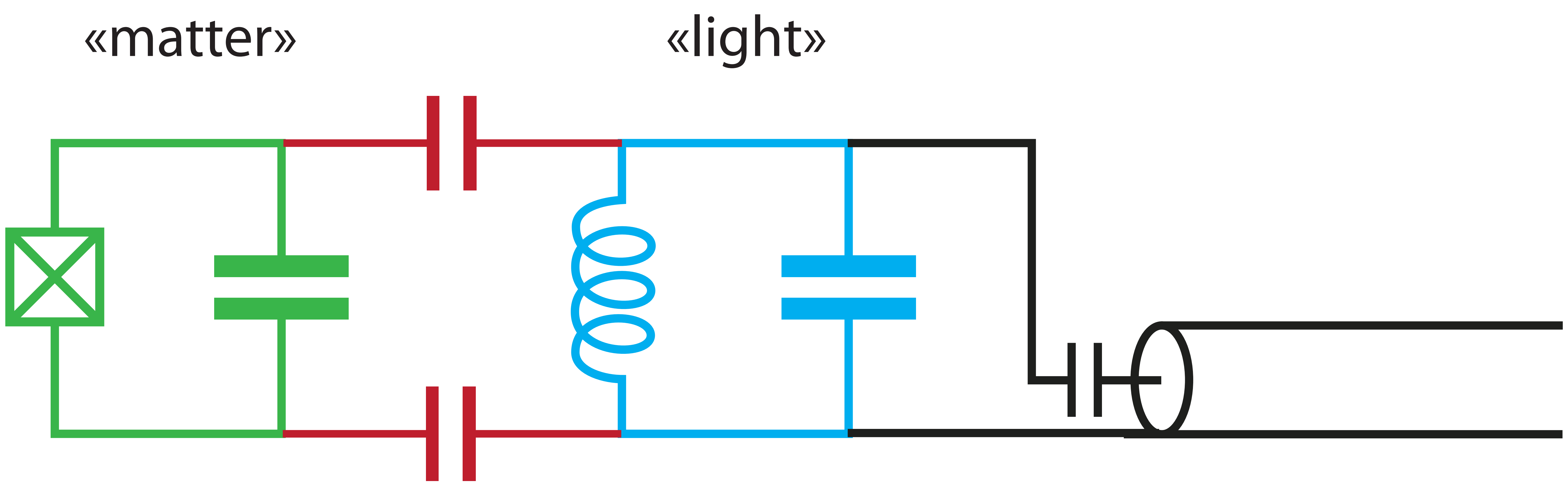
«translation»

$$(P_0(y), \ldots, P_t(y), \ldots, P_T(y))$$

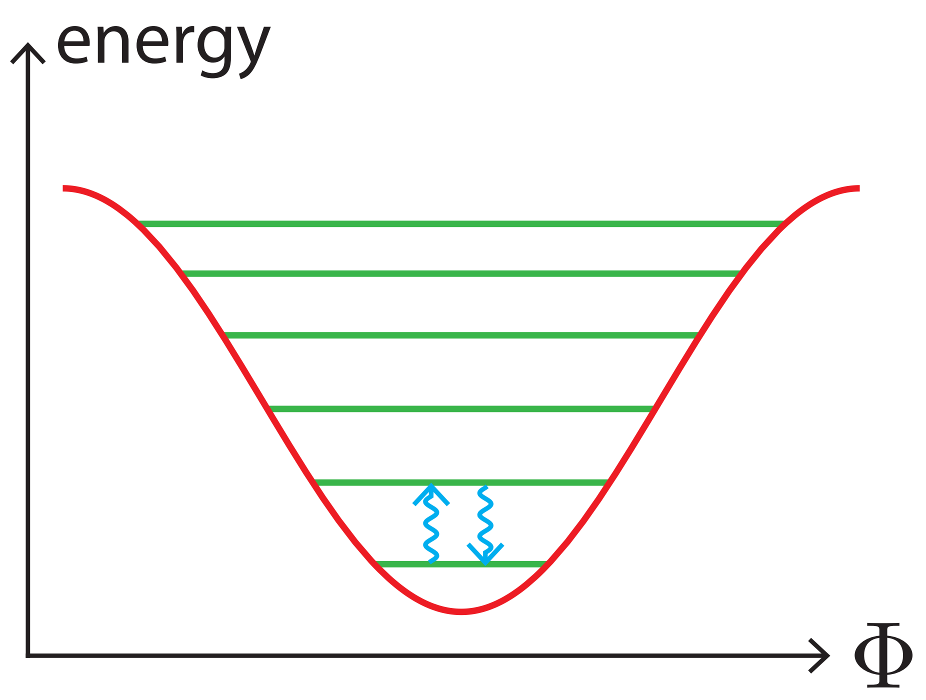
$$\downarrow \qquad \downarrow \qquad \downarrow$$

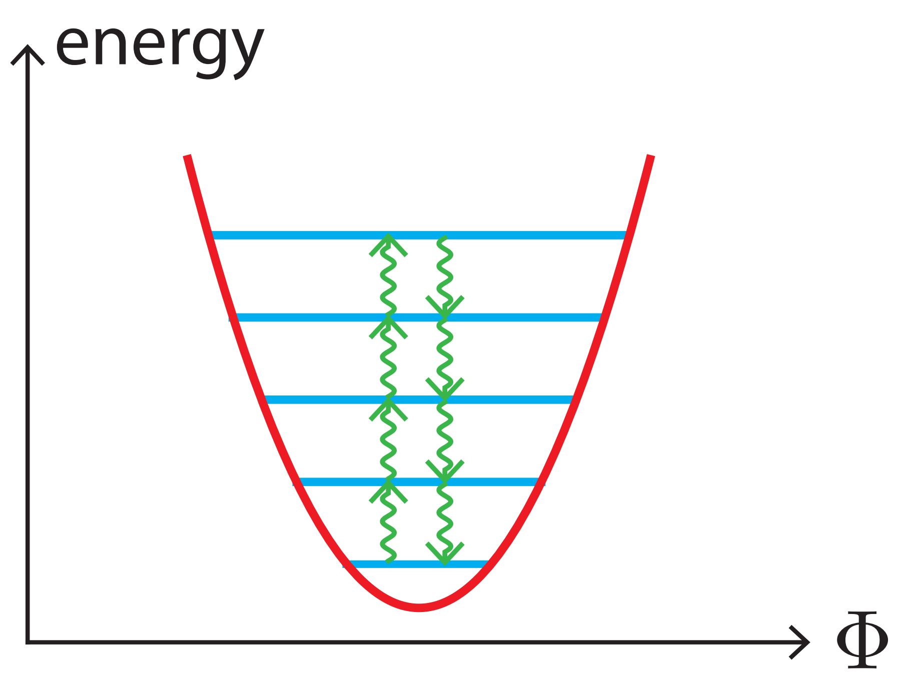
quantum trajectory $(|\psi_0\rangle, \ldots, |\psi_t\rangle, \ldots, |\psi_T\rangle)$

Circuit Quantum Electro-Dynamics

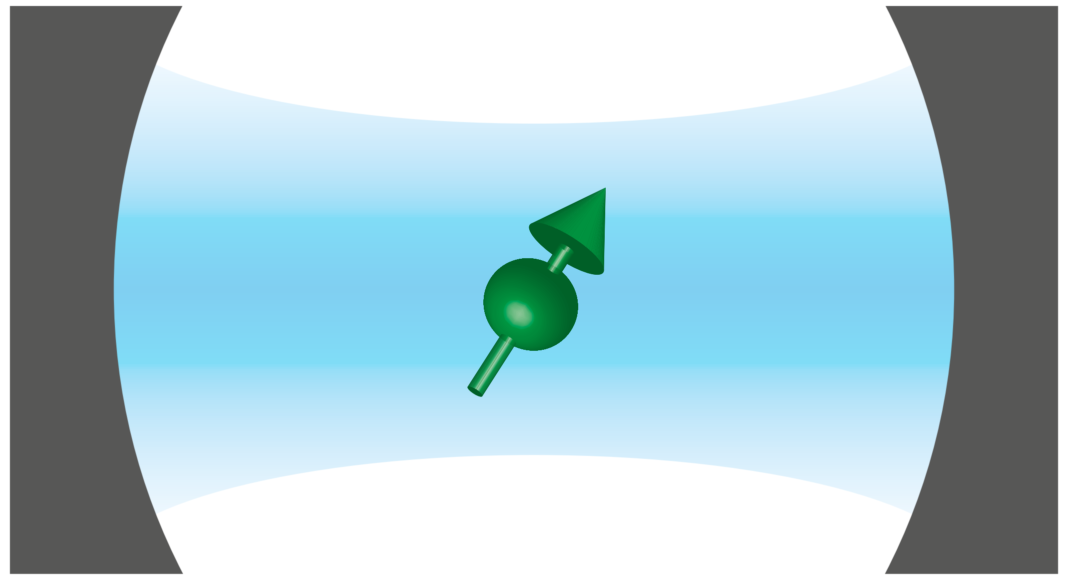


$$H = \frac{Q_1^2}{2C_1} - E_J \cos(\Phi_1) + \frac{Q_2^2}{2C_2} + \frac{\Phi_2^2}{2L_1} + \frac{Q_1 Q_2}{C}$$





Circuit Quantum Electro-Dynamics



$$H = \hbar\omega_q \frac{\hat{\sigma}_z}{2} + \hbar\omega_c \hat{a}^\dagger \hat{a} + \hbar\chi \frac{\hat{\sigma}_z}{2} \hat{a}^\dagger \hat{a}$$

dispersive coupling

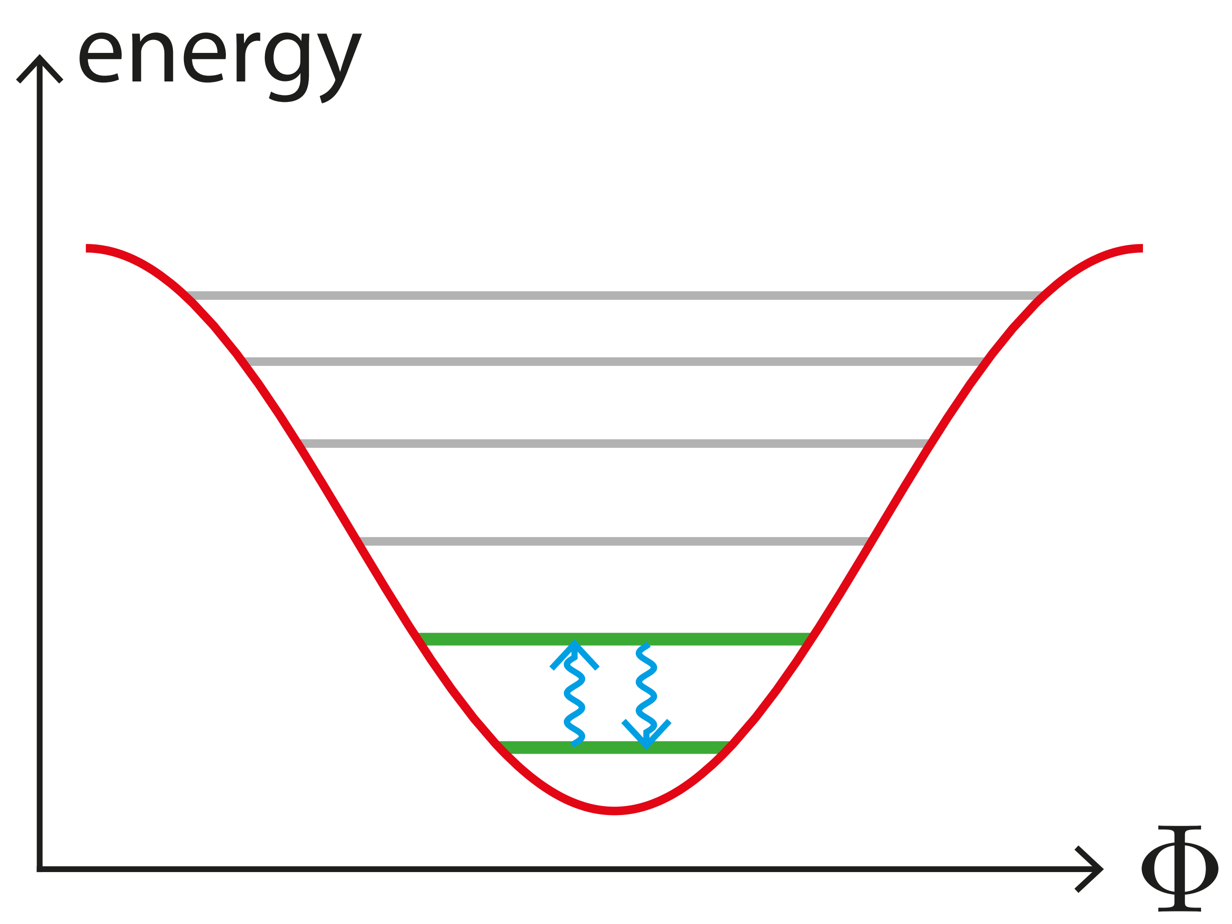
$$|\omega_c - \omega_q| \gg g$$

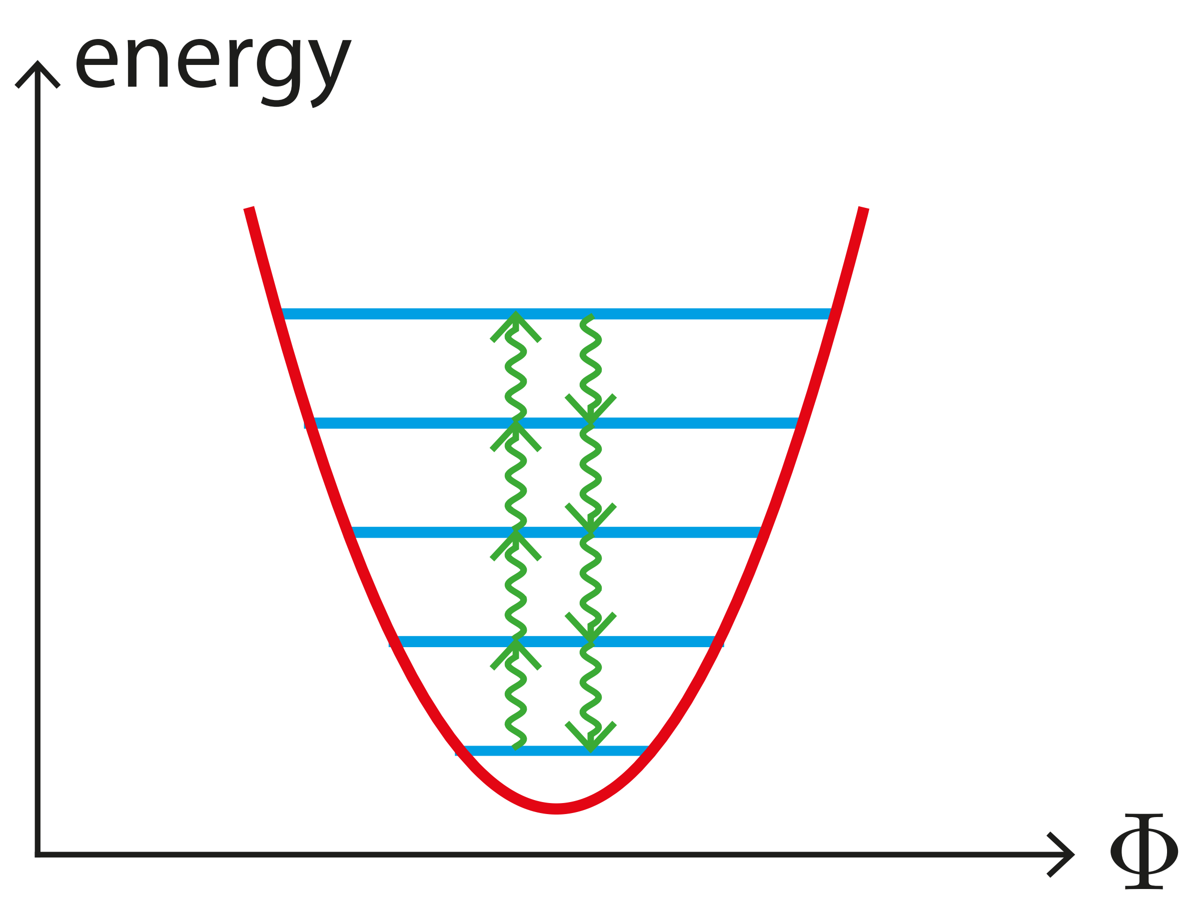
spin orientation

$$\{-\tfrac{1}{2}, \tfrac{1}{2}\}$$

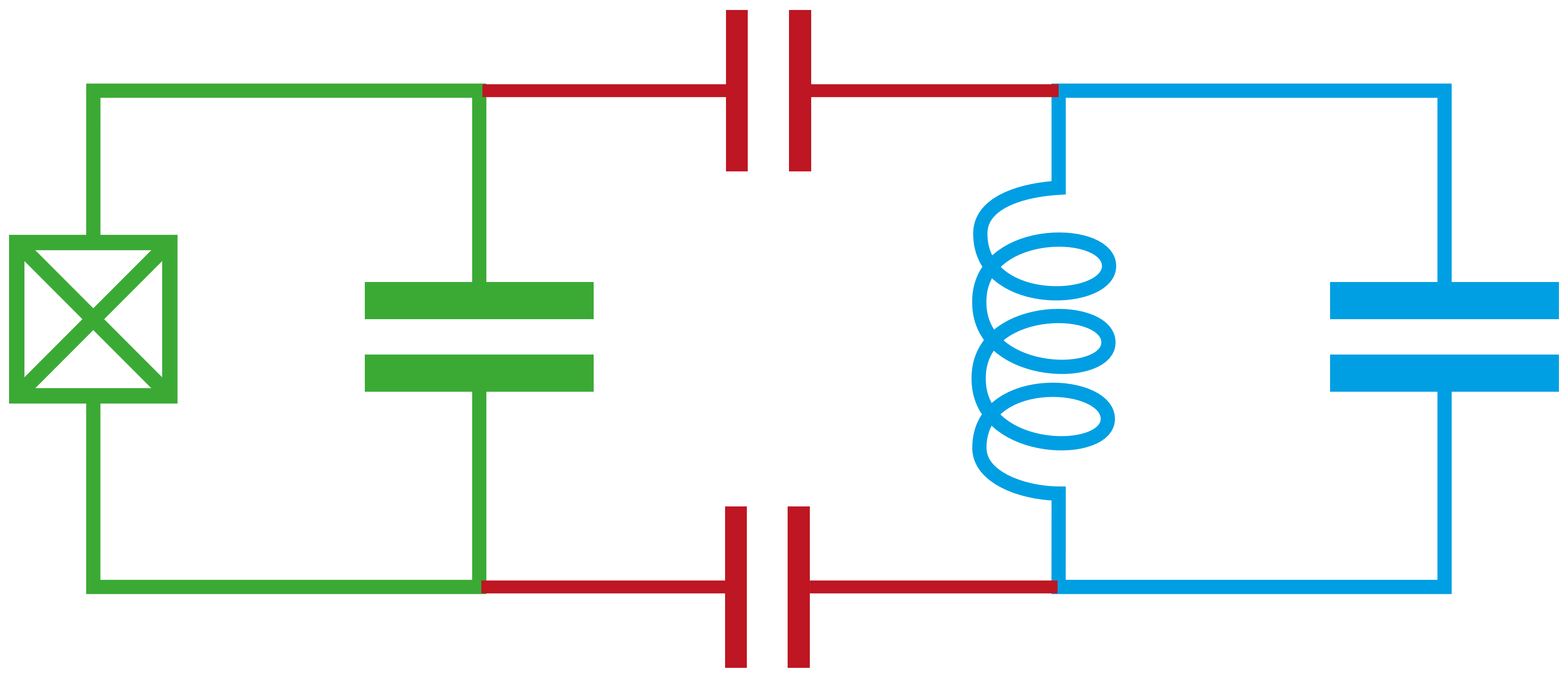
photon number

$$\{0, 1, 2, 3...\}$$



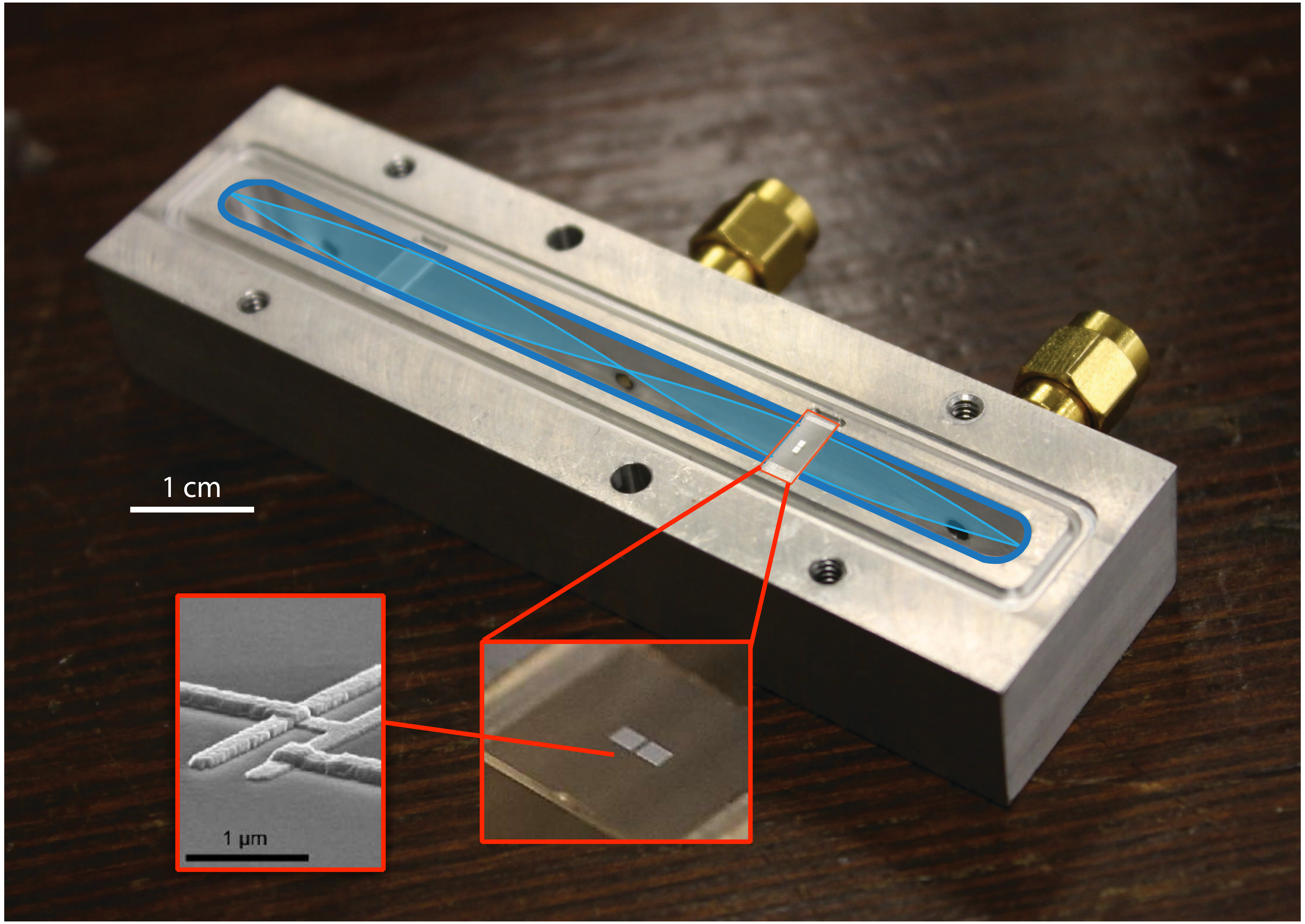


Physical Implementation

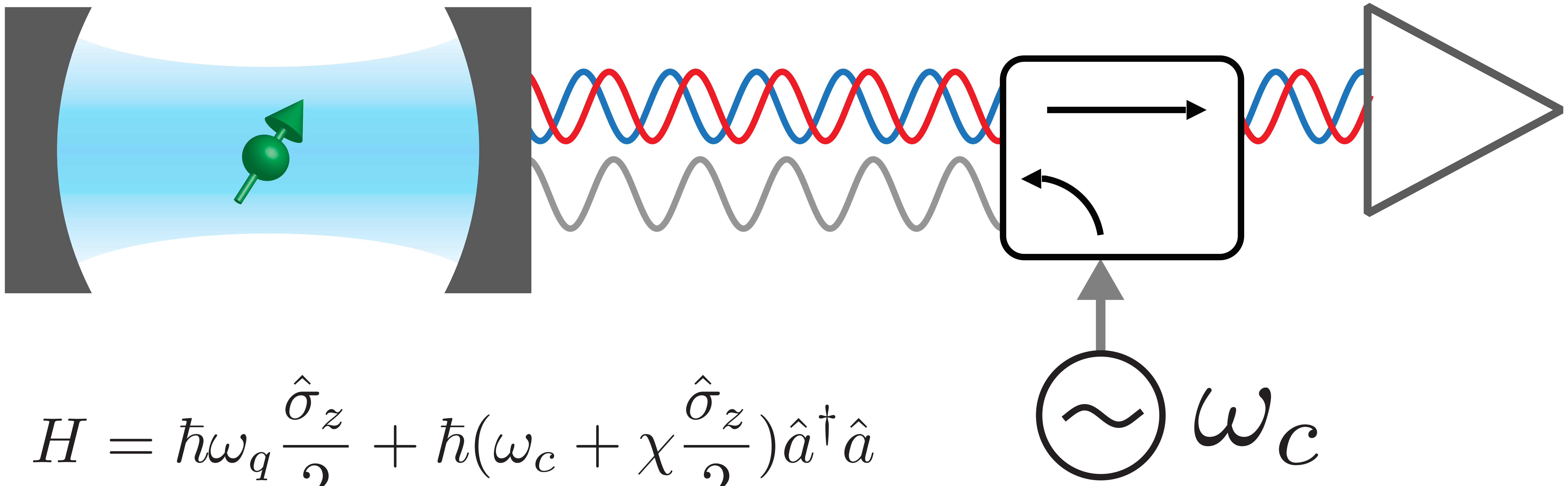


$$\omega_c \sim 2\pi \times 7 \text{ GHz}$$
$$\omega_q \sim 2\pi \times 5 \text{ GHz}$$

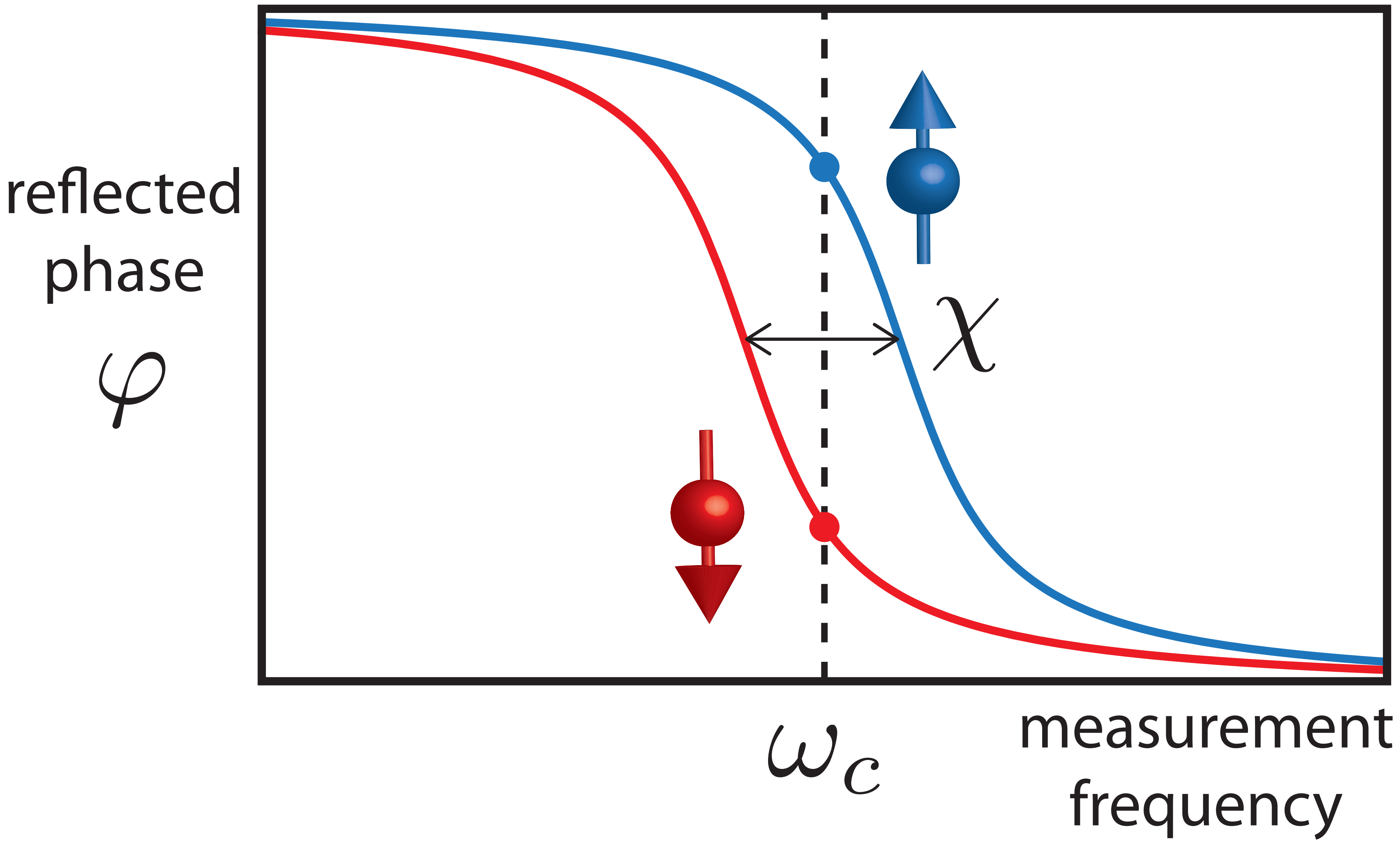
$$\chi \sim 2\pi \times 0.3 \text{ MHz}$$

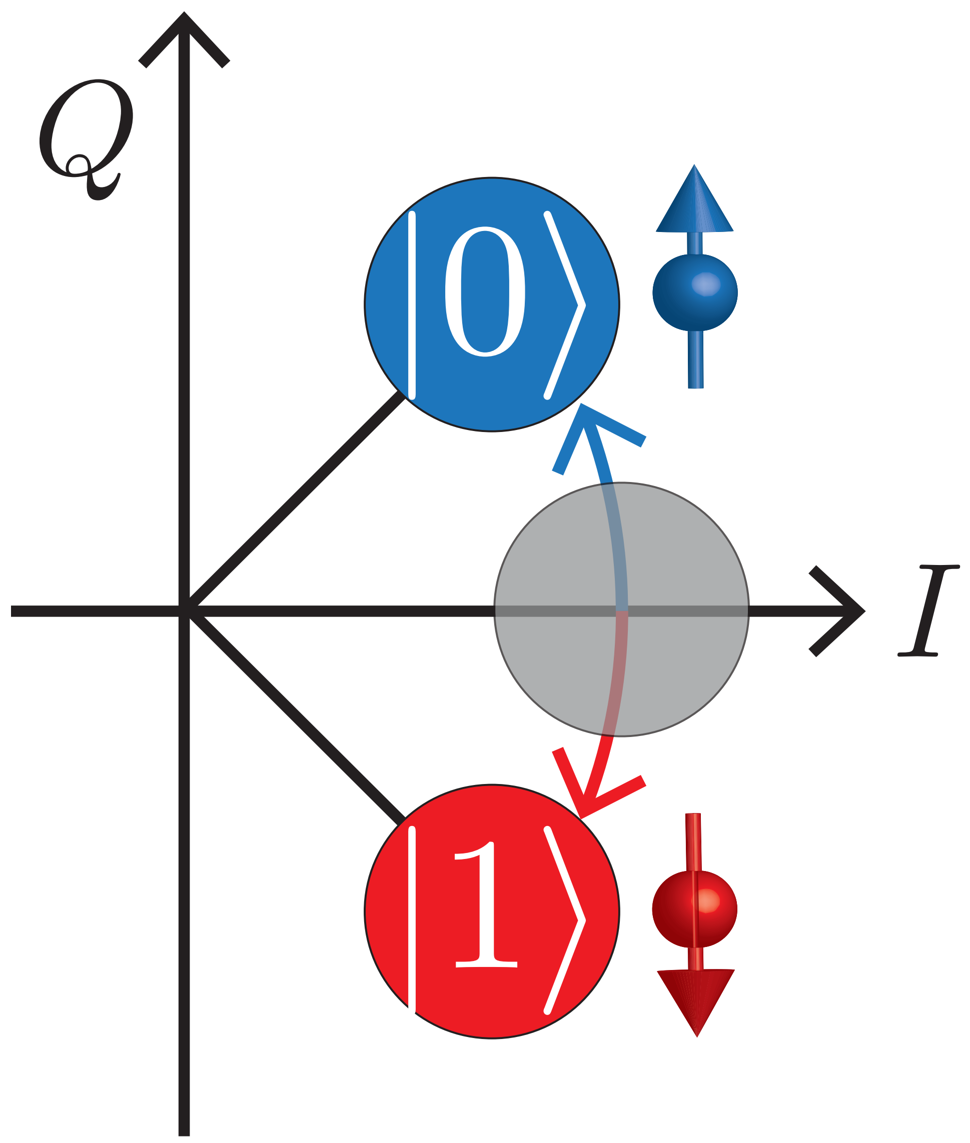


Dispersive Measurement



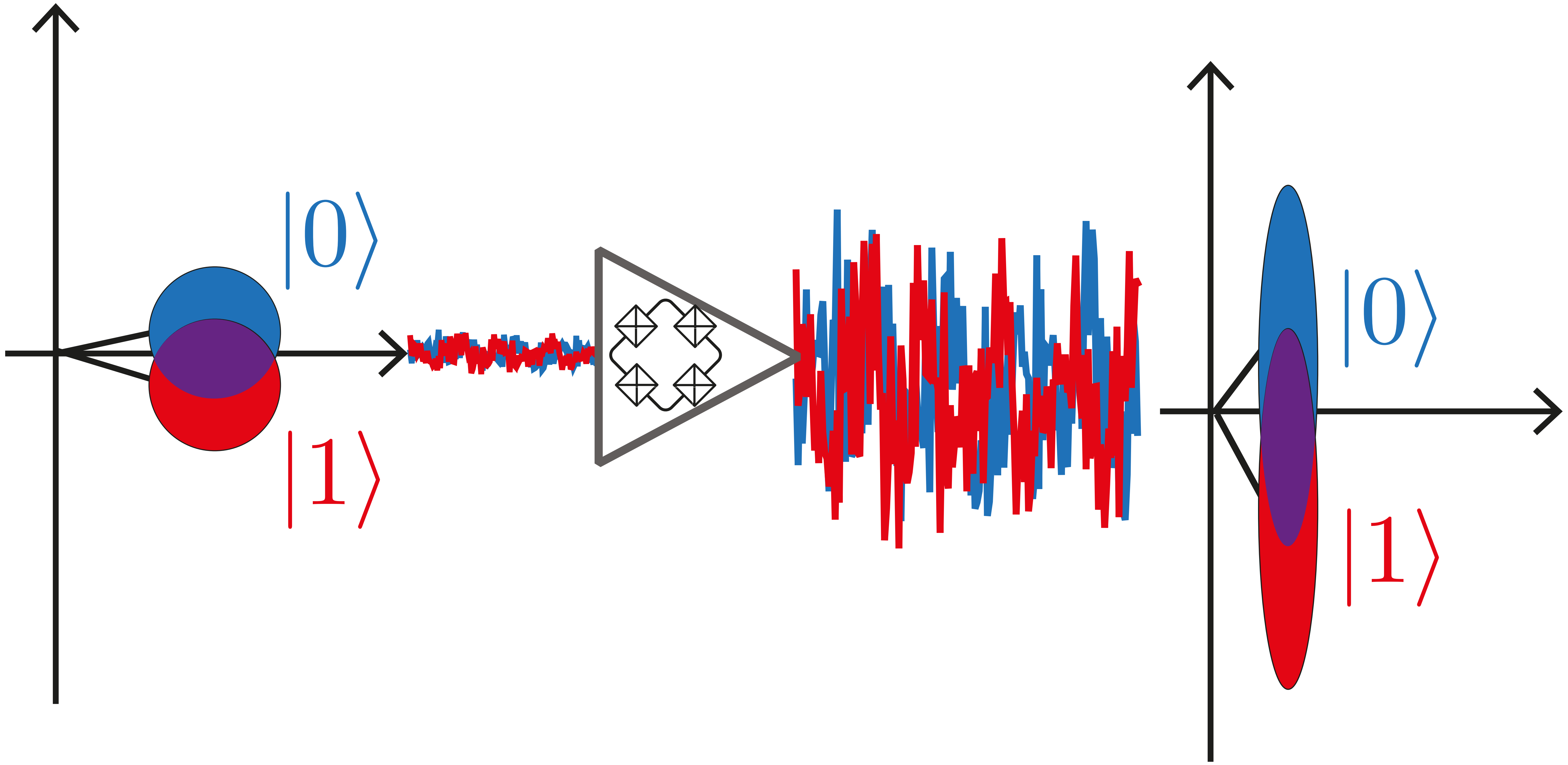
$$H = \hbar\omega_q \frac{\hat{\sigma}_z}{2} + \hbar\underbrace{(\omega_c + \chi\frac{\hat{\sigma}_z}{2})}_{\text{effective frequency}}\hat{a}^\dagger\hat{a}$$

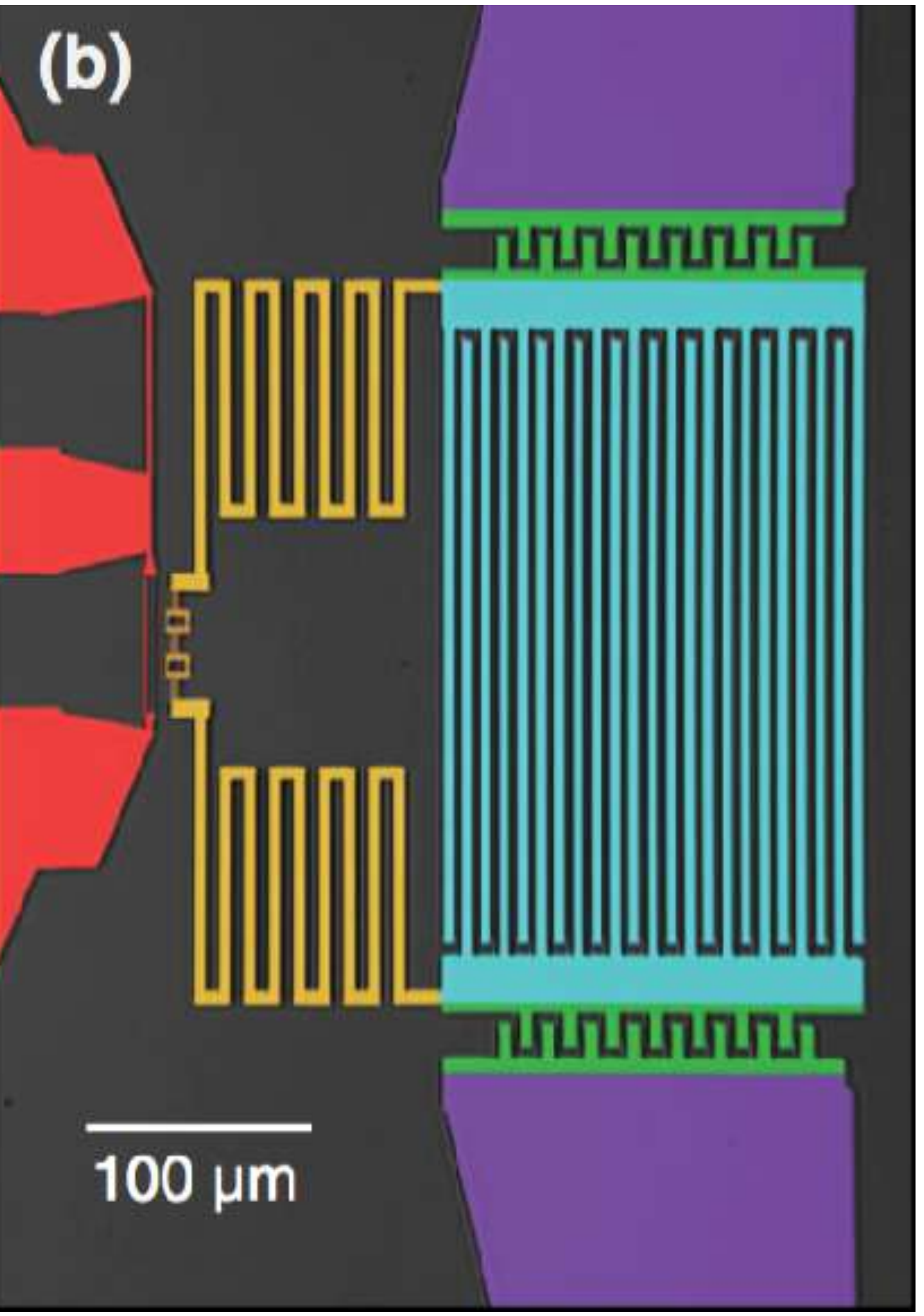




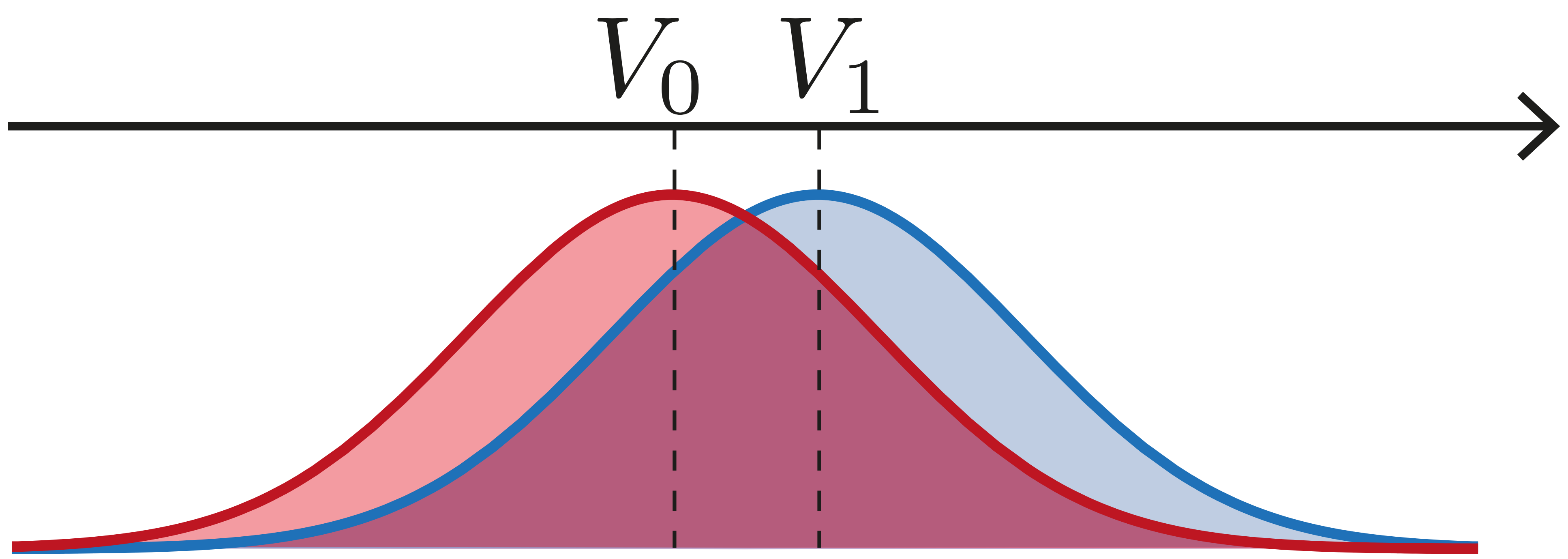
Experiment

Josephson Amplifier

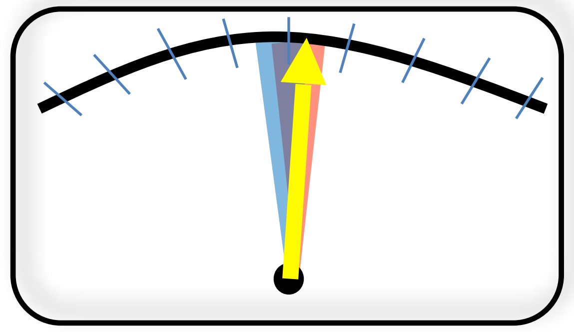




average measurement outcome

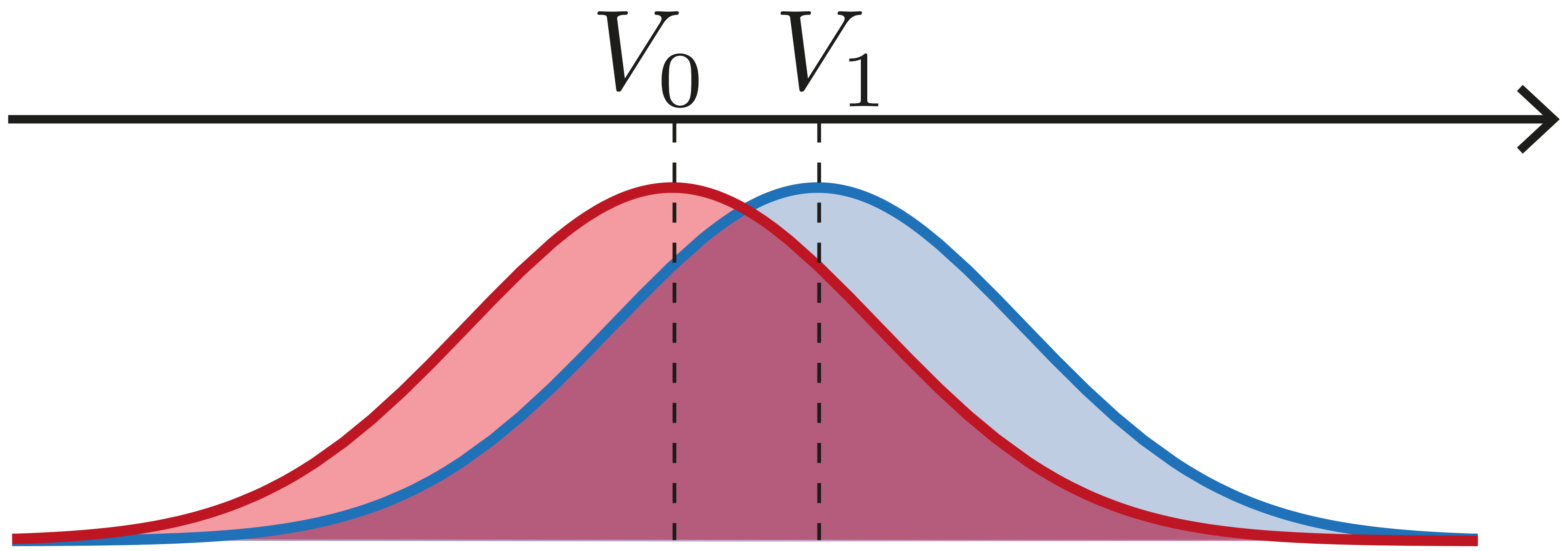


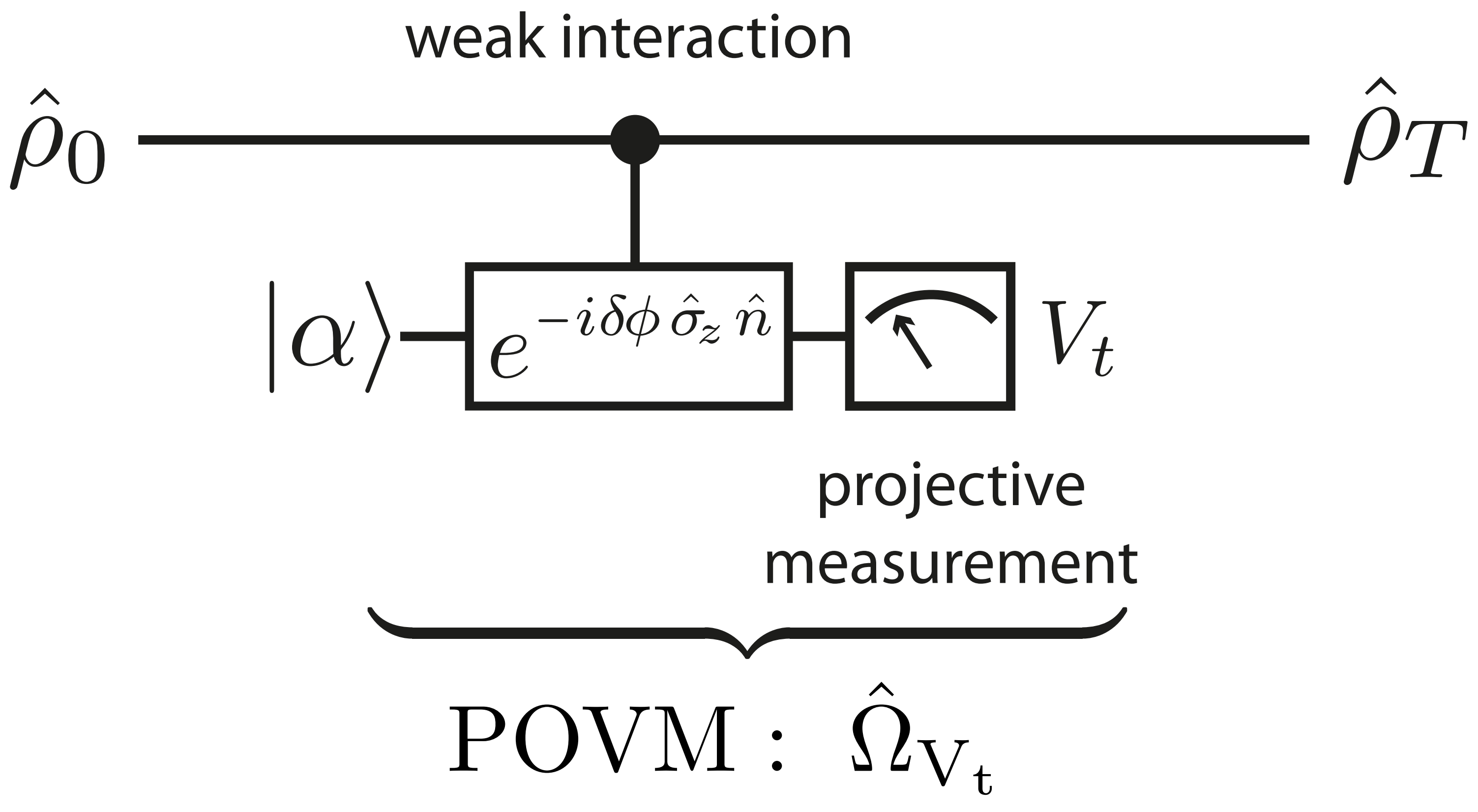
Strong vs Weak Measurement



Weak

average measurement outcome

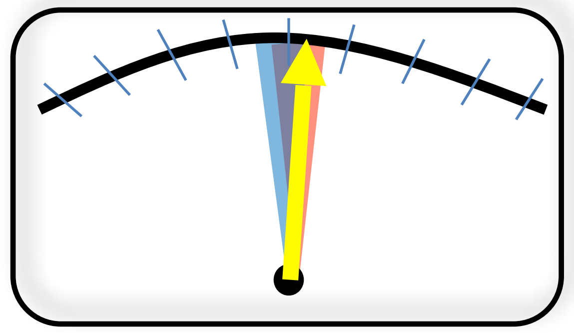




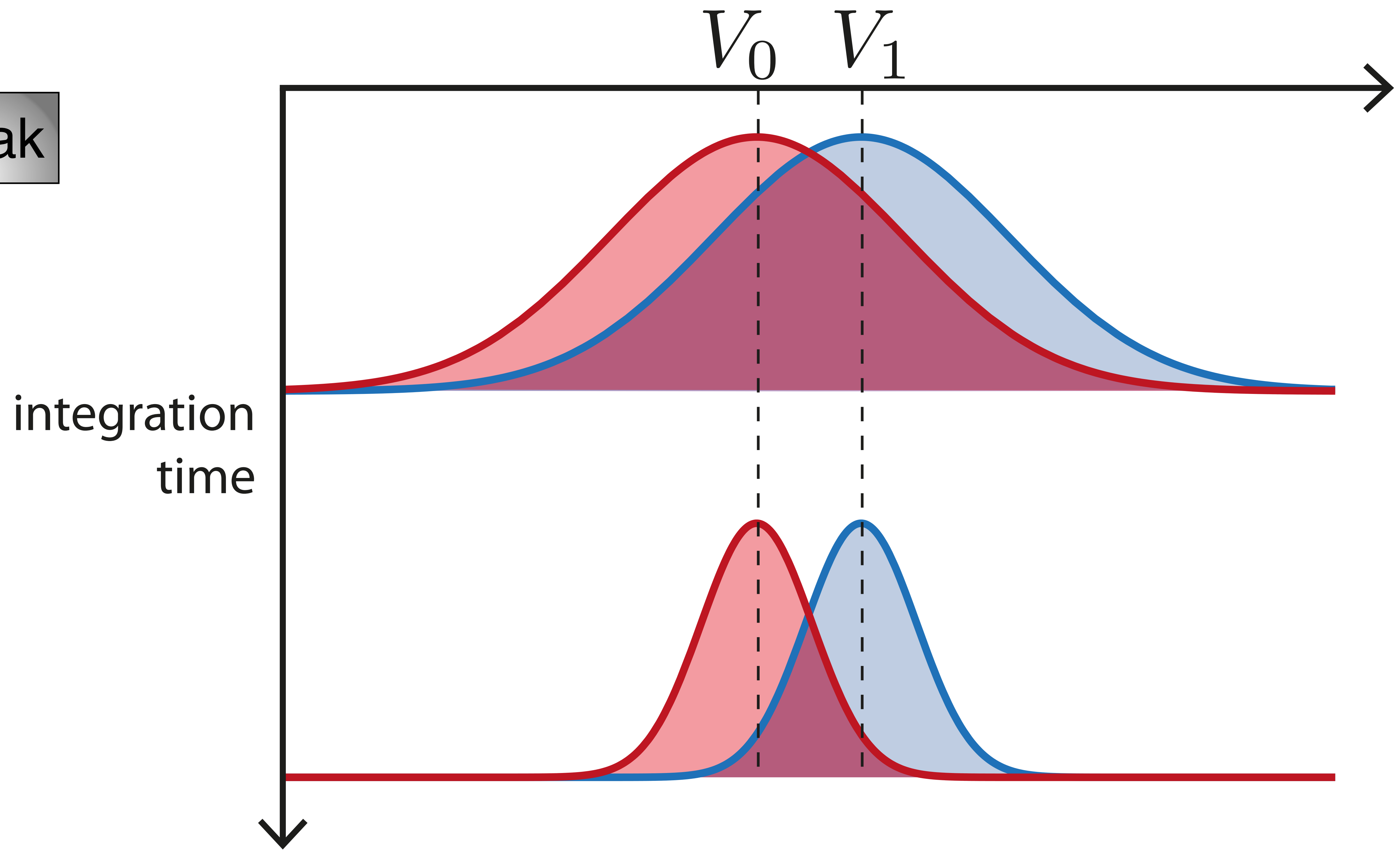
« Partial-Measurement Backaction and Nonclassical Weak Values in a Superconducting Circuit »
Groen et al., PRL 2013 (Delft)

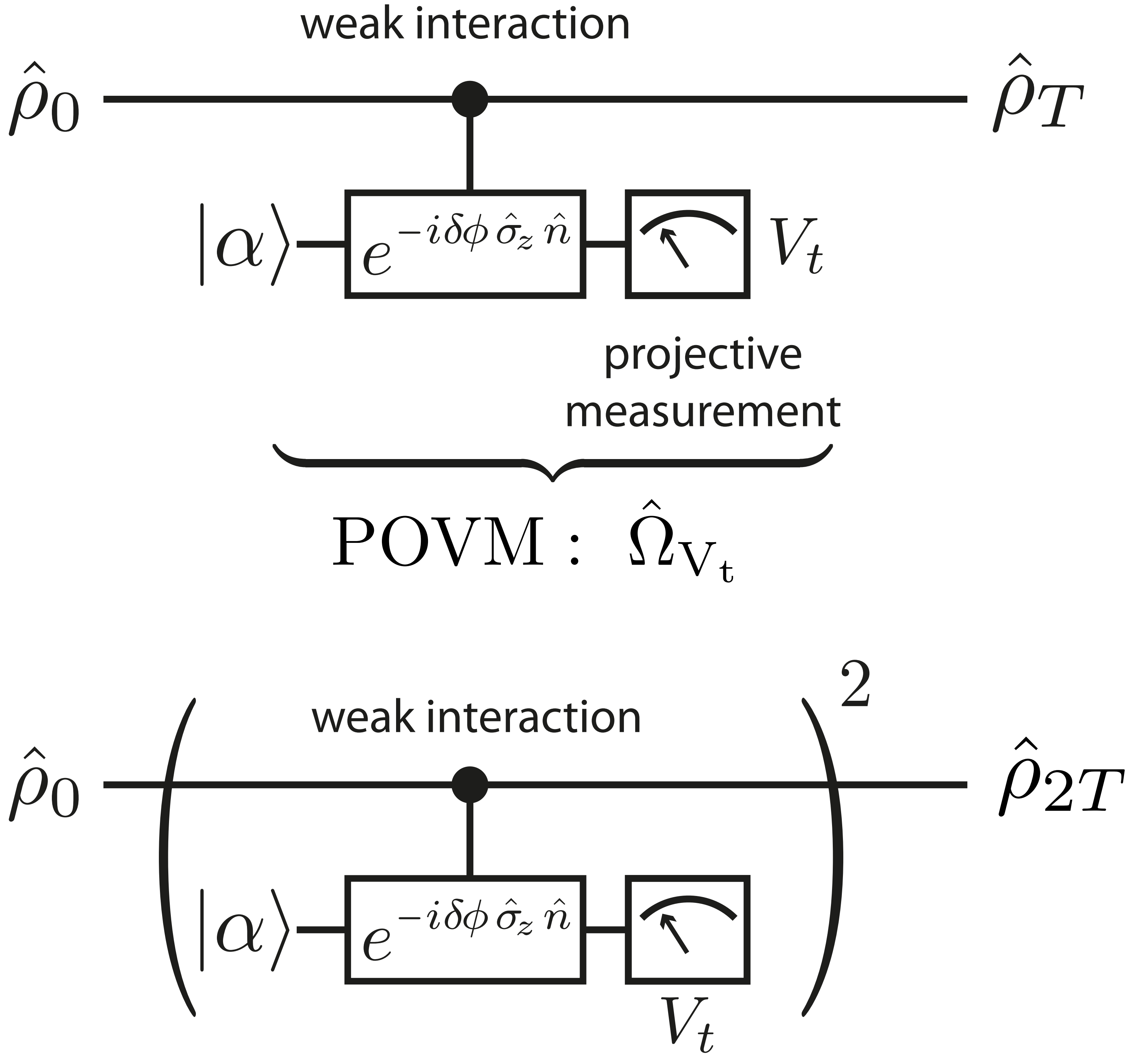
Trajectories

Strong vs Weak Measurement



Weak



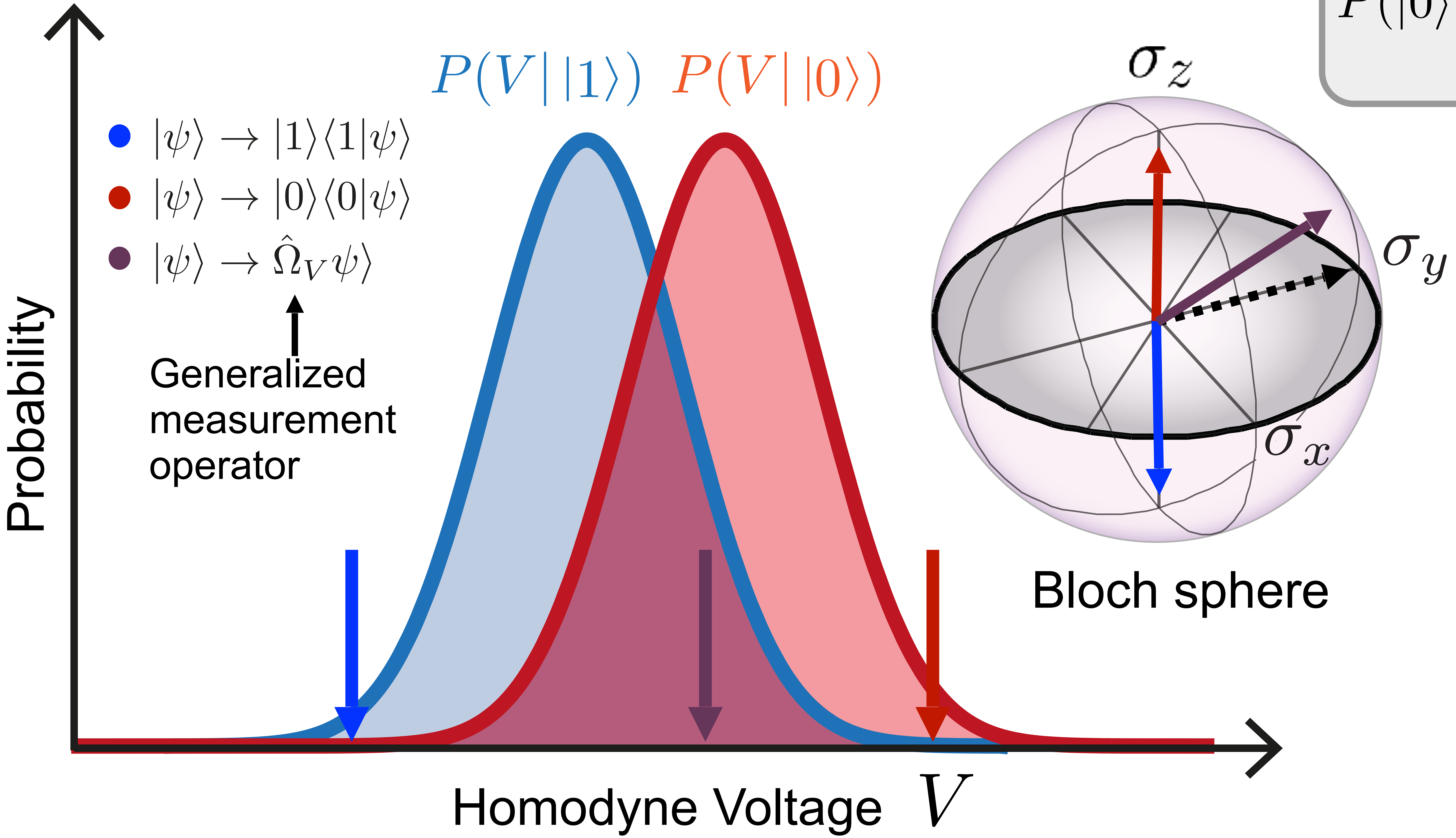


Trajectories

Strong vs Weak Measurement

Quantum Trajectories

What does the detector signal tell us?

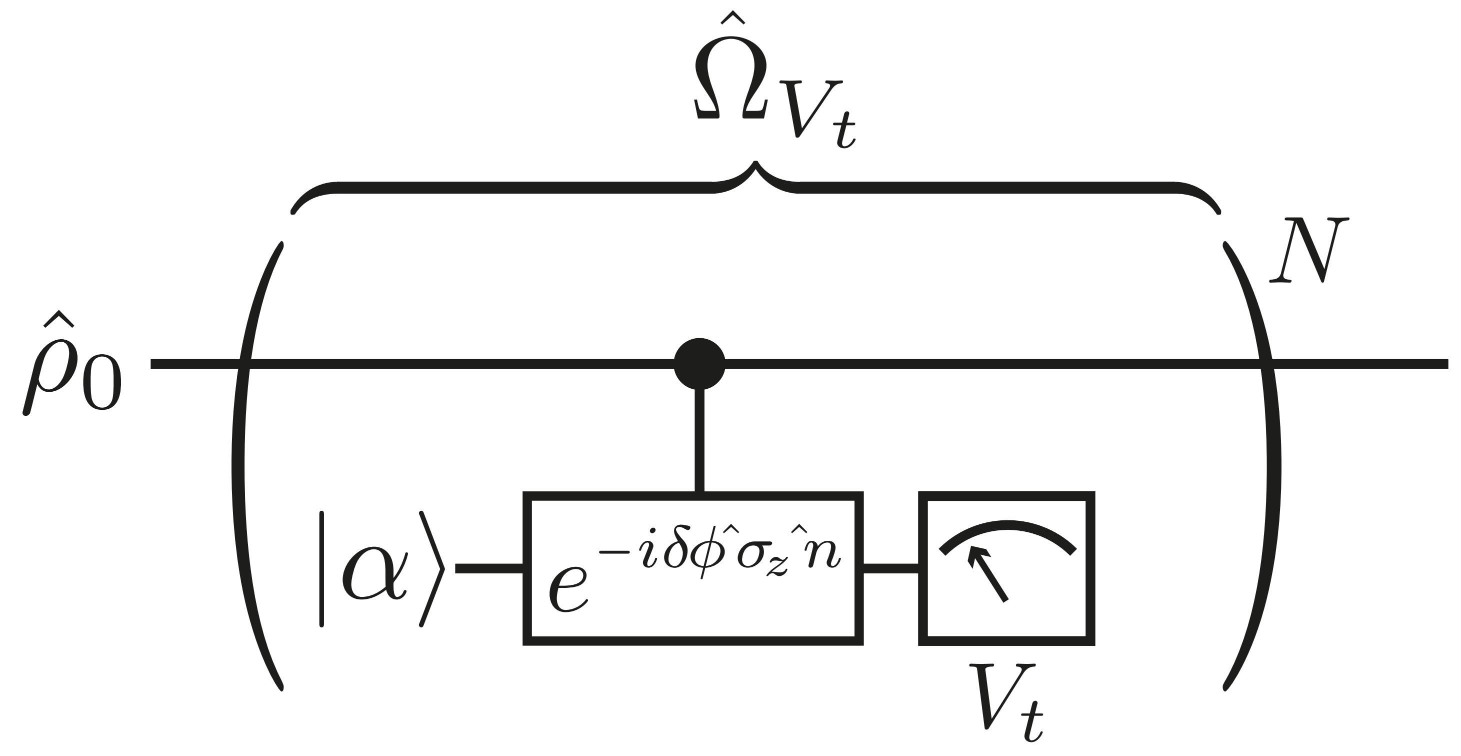


Bloch sphere

Bayesian inference

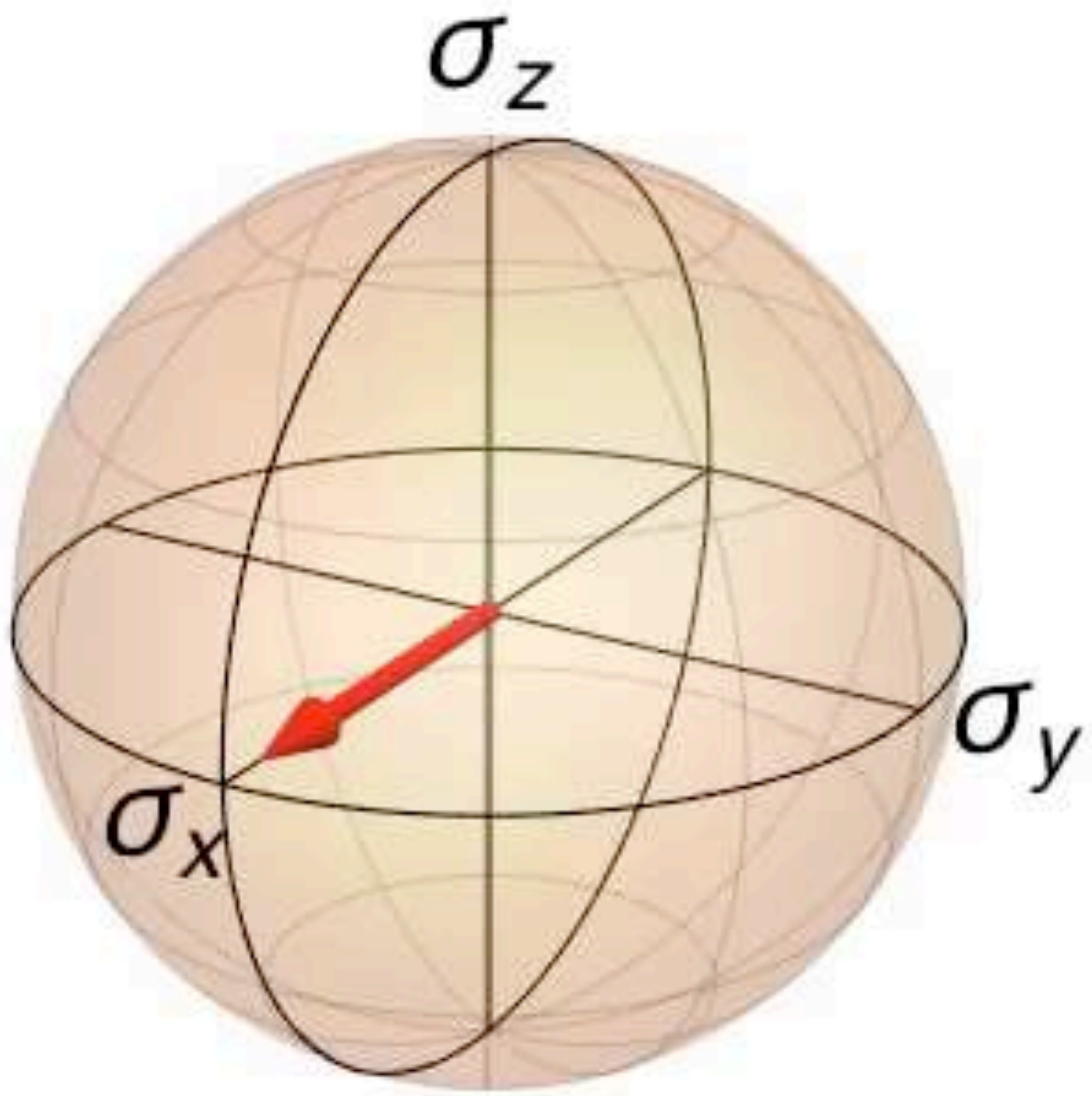
$$P(|0\rangle|V) = \frac{P(V|\,|0\rangle)P(|0\rangle)}{P(V)}$$

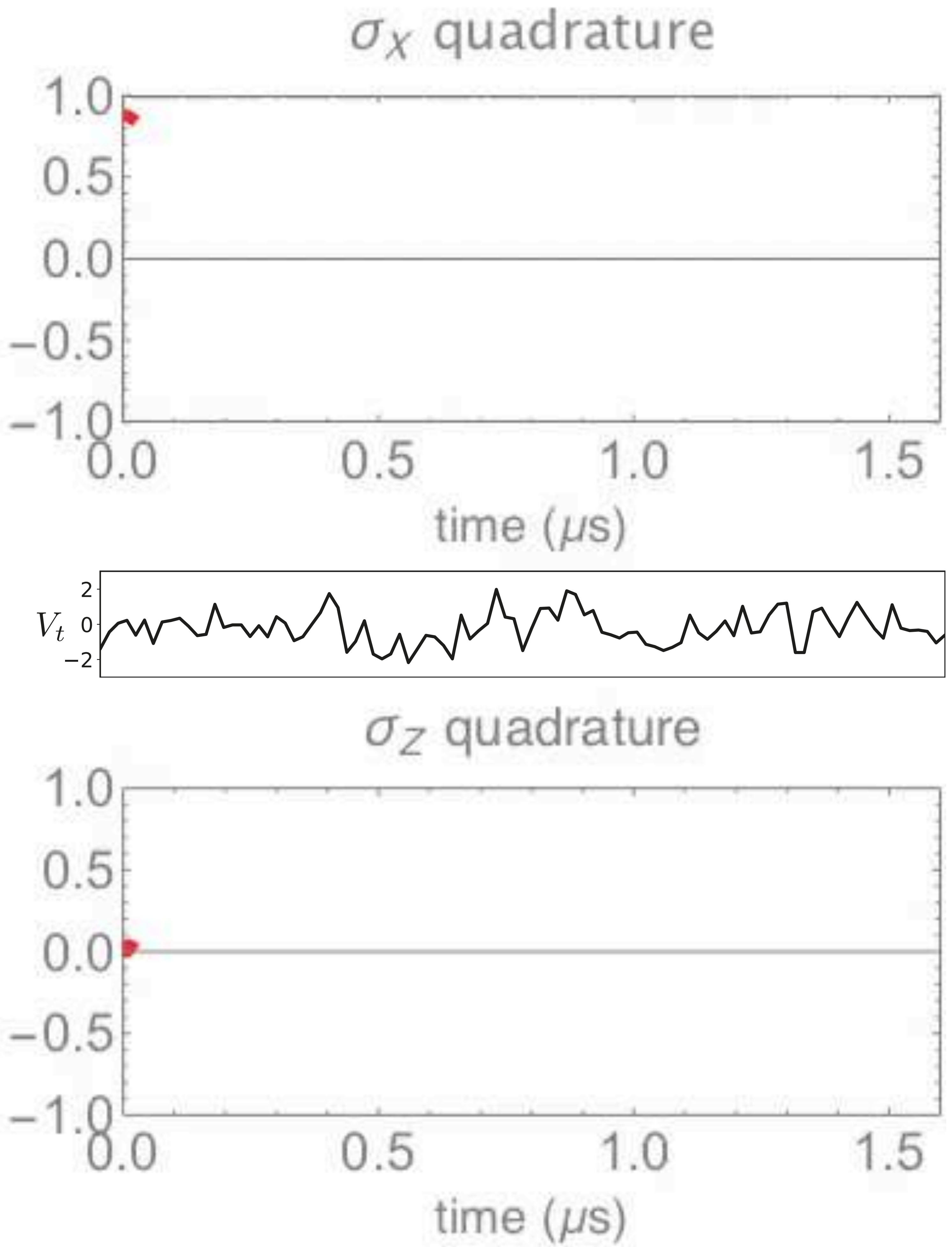
Quantum Trajectories



$$\hat{\rho}_T = \frac{\hat{\Omega}_{V_T}\dots\hat{\Omega}_{V_t}\dots\hat{\Omega}_{V}\;\hat{A}\;\hat{\rho}_0\,\hat{A}^{\dagger}\hat{\Omega}^{\dagger}_{V_0}\dots\hat{\Omega}^{\dagger}_{V_t}\dots\hat{\Omega}^{\dagger}_{V_T}}{\mathrm{Tr}(\hat{\Omega}_{V_T}\dots\hat{\Omega}_{V_t}\dots\hat{\Omega}_{V}\;\hat{A}\;\hat{\rho}_0\,\hat{A}^{\dagger}\hat{\Omega}^{\dagger}_{V_0}\dots\hat{\Omega}^{\dagger}_{V_t}\dots\hat{\Omega}^{\dagger}_{V_T})}$$

Experimental data:

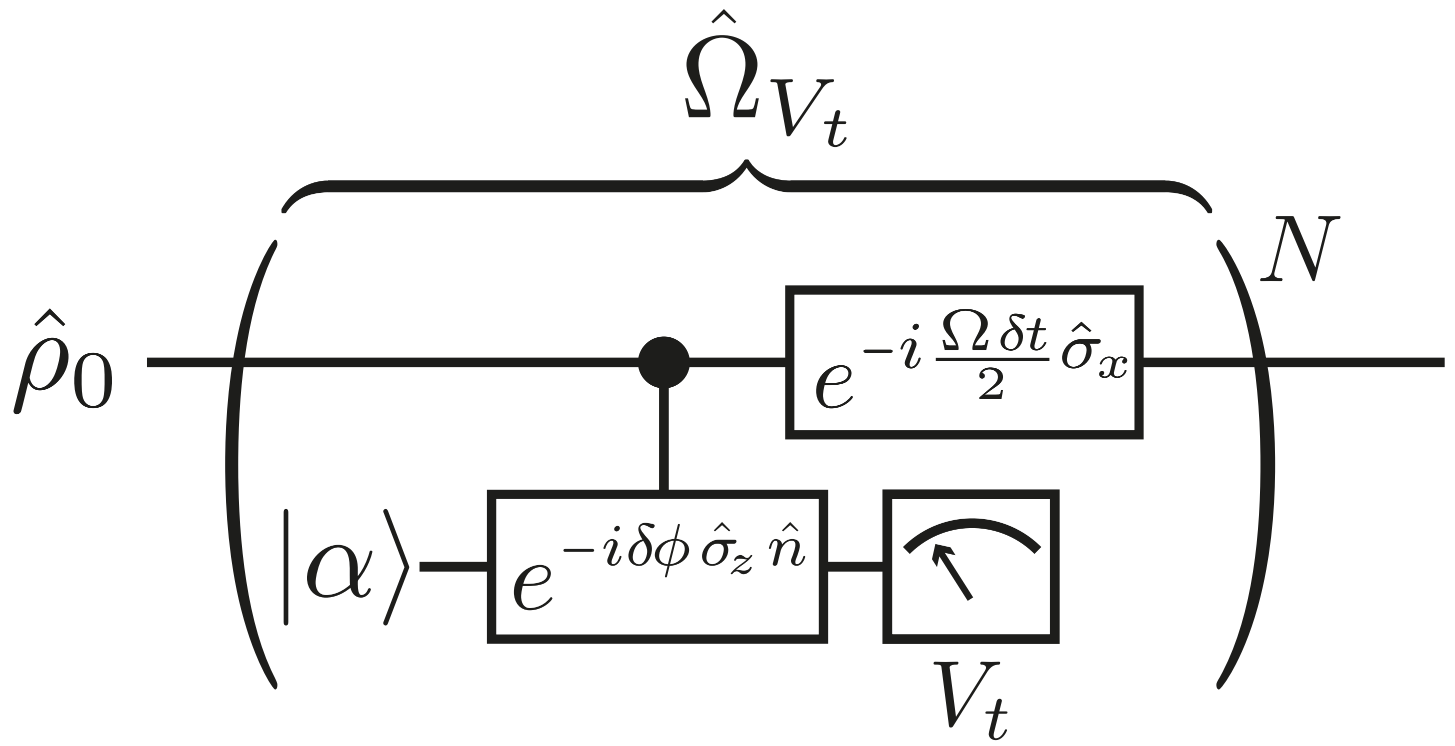




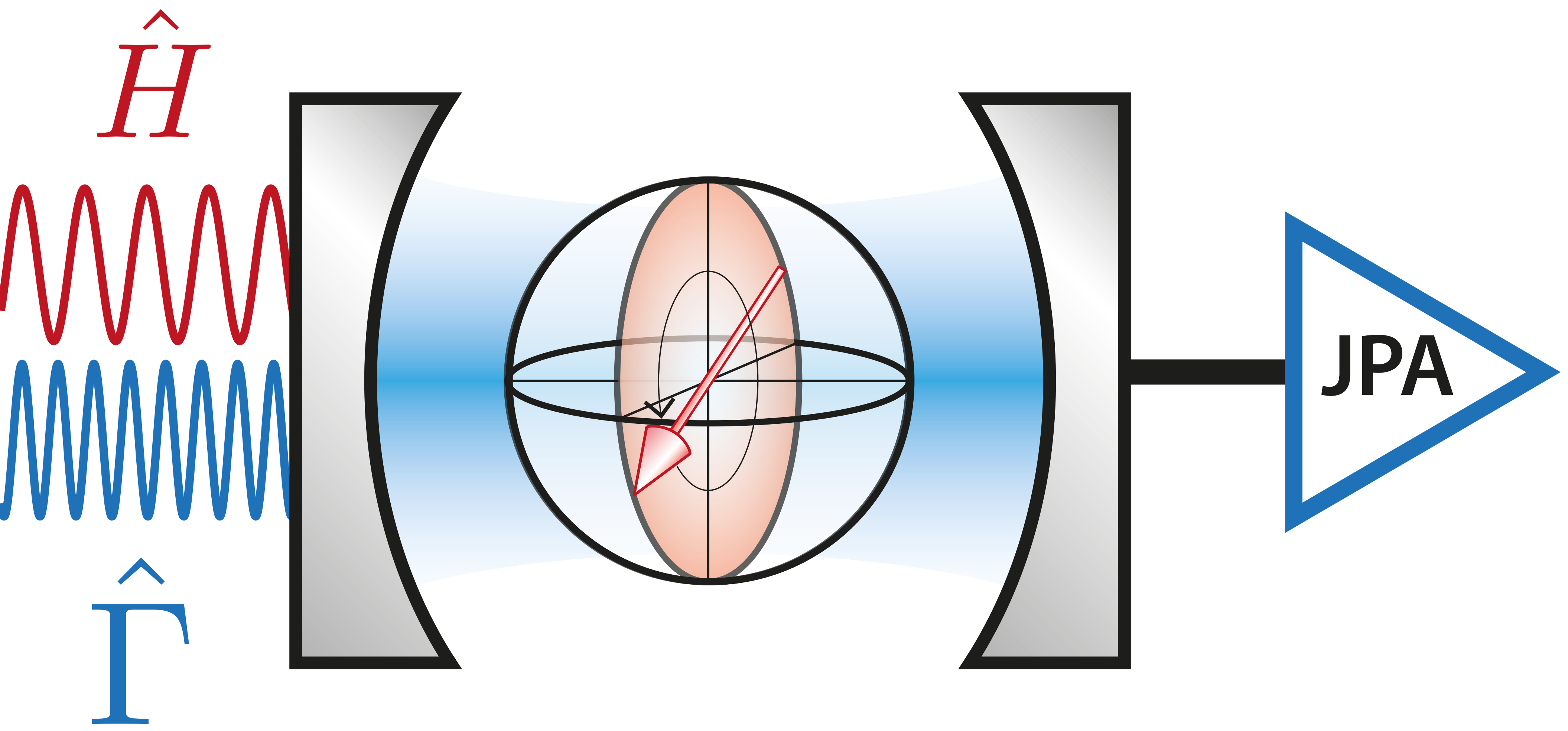
K. Murch *et al.*, Nature **502** 211 (2013).

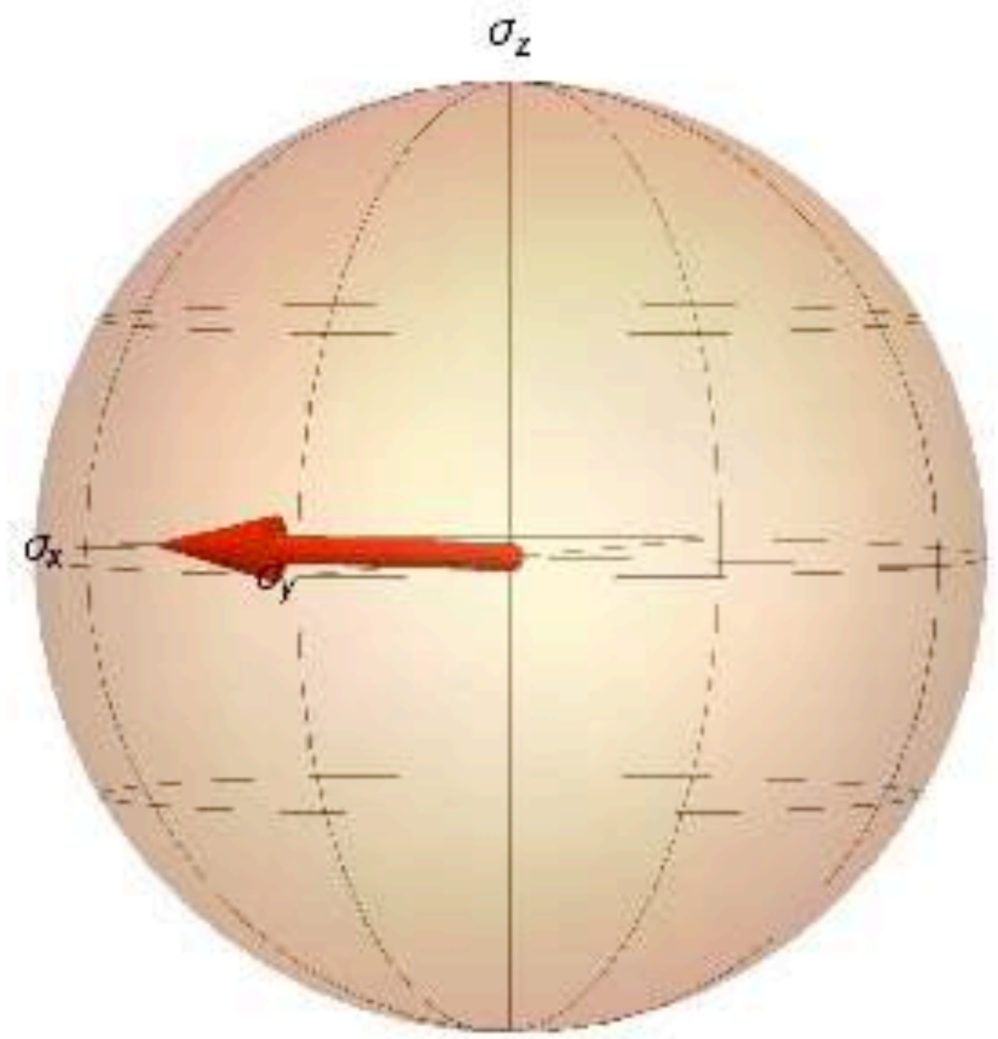
Trajectories

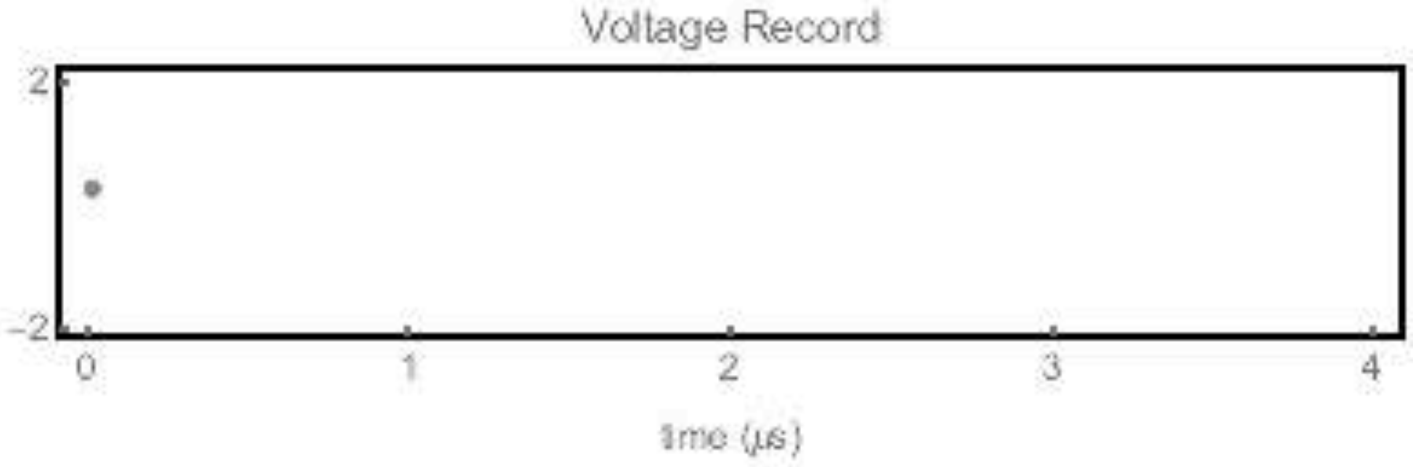
Weak Measurement of a Driven Qubit

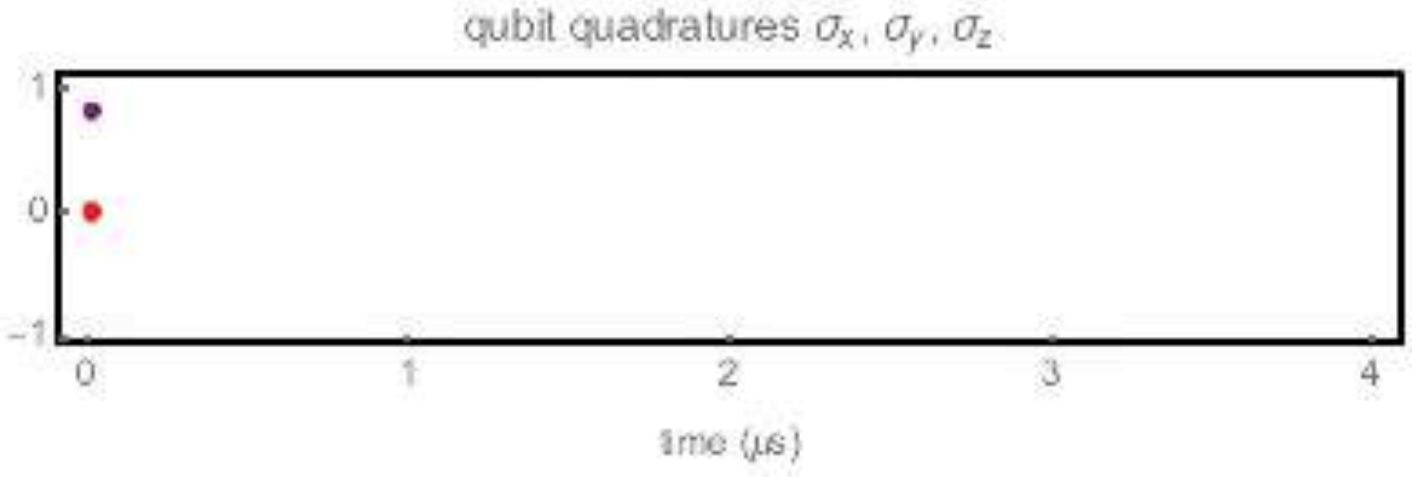


$$\hat{\rho}_T = \frac{\hat{\Omega}_{V_T}\ldots\hat{\Omega}_{V_t}\ldots\hat{\Omega}_{V}\;\hat{A}\;\hat{\rho}_0\,\hat{A}^{\dagger}\hat{\Omega}^{\dagger}_{V_0}\ldots\hat{\Omega}^{\dagger}_{V_t}\ldots\hat{\Omega}^{\dagger}_{V_T}}{\mathrm{Tr}(\hat{\Omega}_{V_T}\ldots\hat{\Omega}_{V_t}\ldots\hat{\Omega}_{V}\;\hat{A}\;\hat{\rho}_0\,\hat{A}^{\dagger}\hat{\Omega}^{\dagger}_{V_0}\ldots\hat{\Omega}^{\dagger}_{V_t}\ldots\hat{\Omega}^{\dagger}_{V_T})}$$





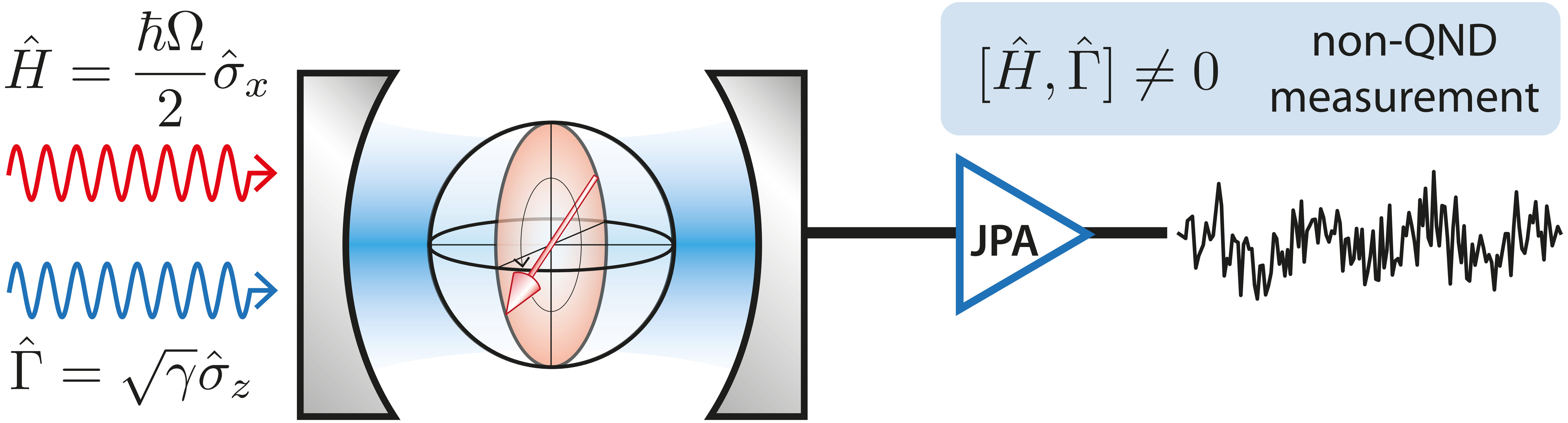




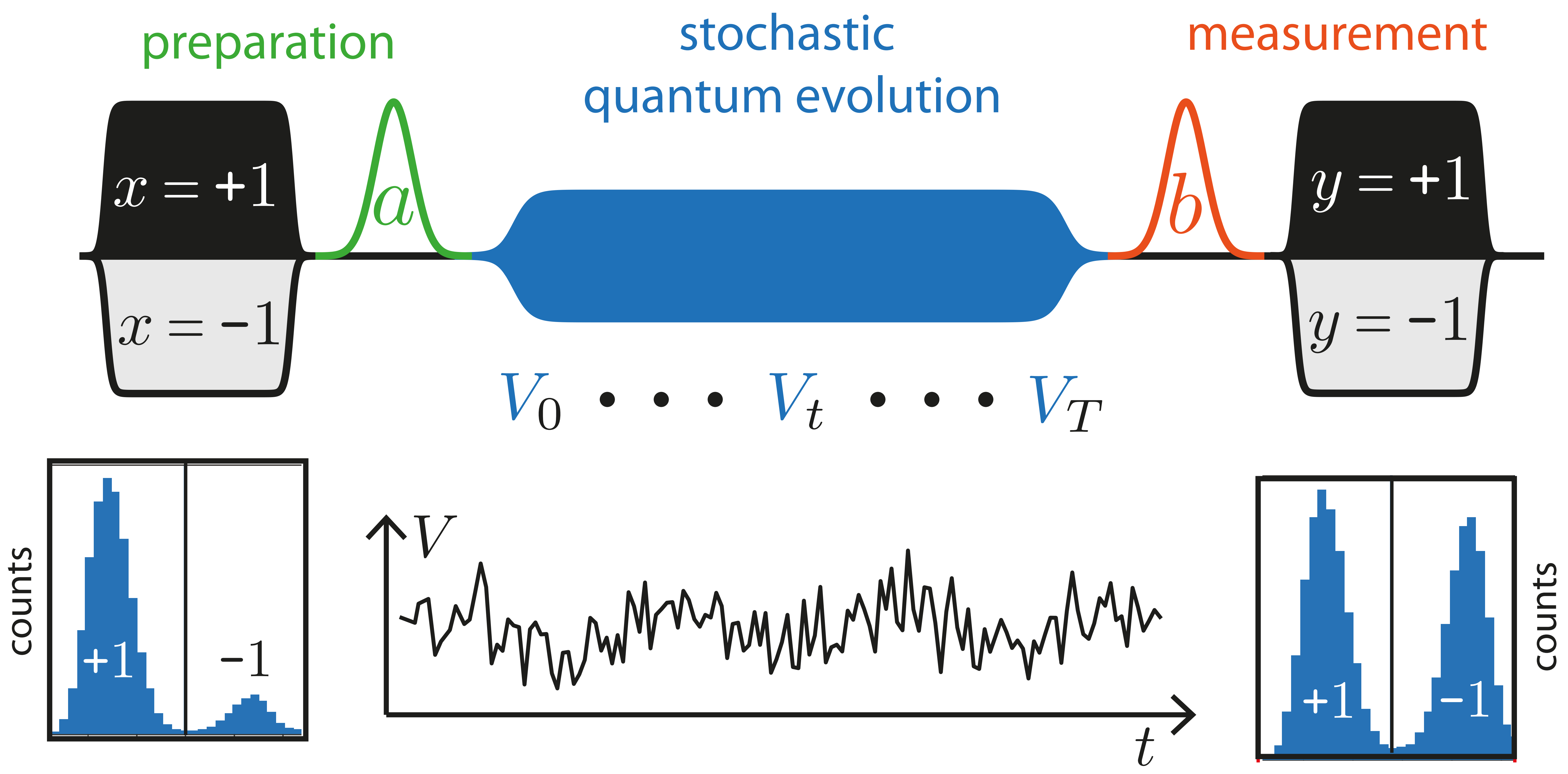
$$[\hat{H}, \hat{\Gamma}] \neq 0$$

Trajectories

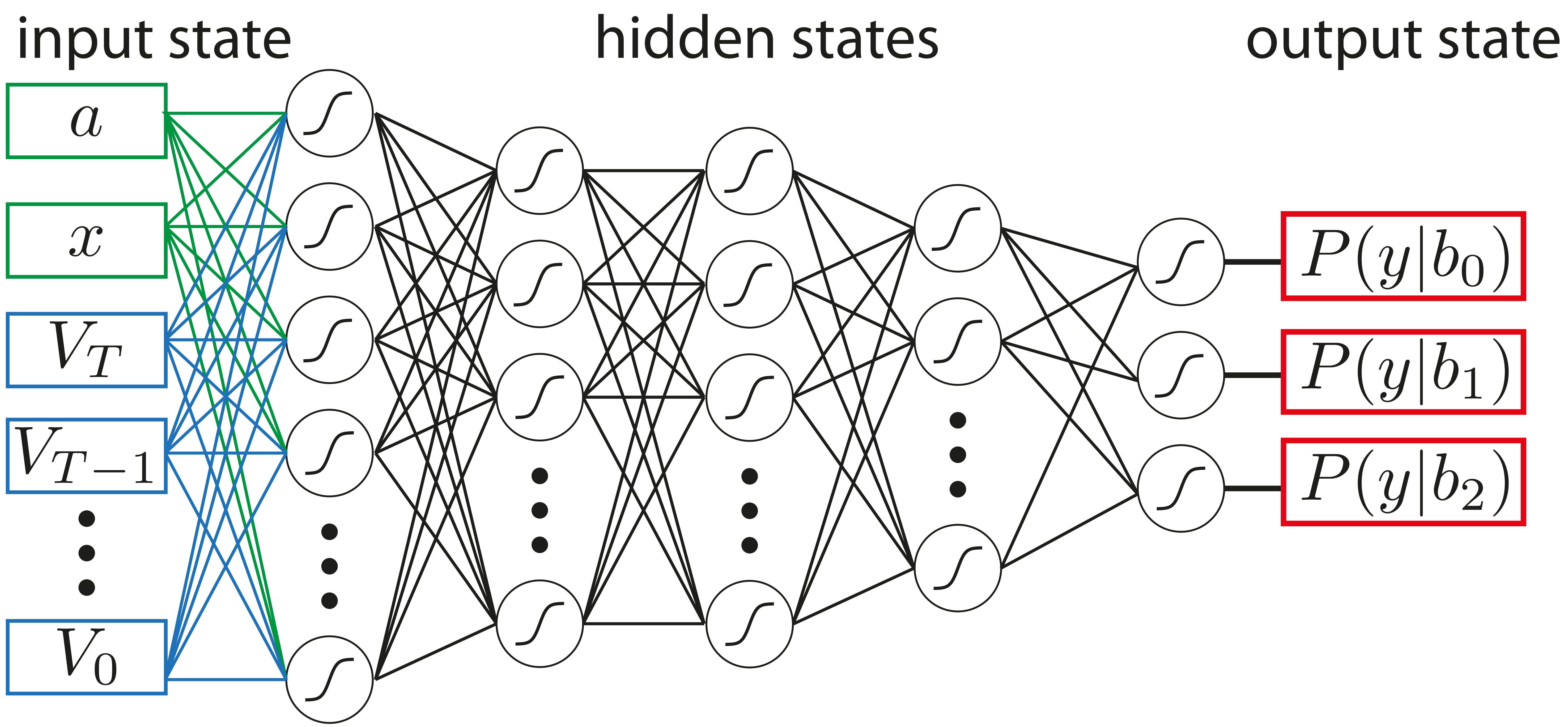
Weak Measurement of a Driven Qubit



Experiment
1.5 millions repetitions
at a rate of **0.5 ms**



Deep Neural Network



$$\boxed{\text{input}} \longrightarrow \vec{h}_0 \xrightarrow[W_1]{\sigma} \vec{h}_1 \xrightarrow[W_2]{\sigma} \vec{h}_2 \xrightarrow[W_3]{\sigma} \vec{h}_3 \xrightarrow[W_4]{\sigma} \vec{h}_4 \longrightarrow \boxed{\text{output}}$$

each layer is represented as a vector of neurones

weight matrix connecting each layer

neurones biases

$$\vec{h}_{n+1} = \sigma(W_n.\vec{h}_n + \vec{b}_n)$$

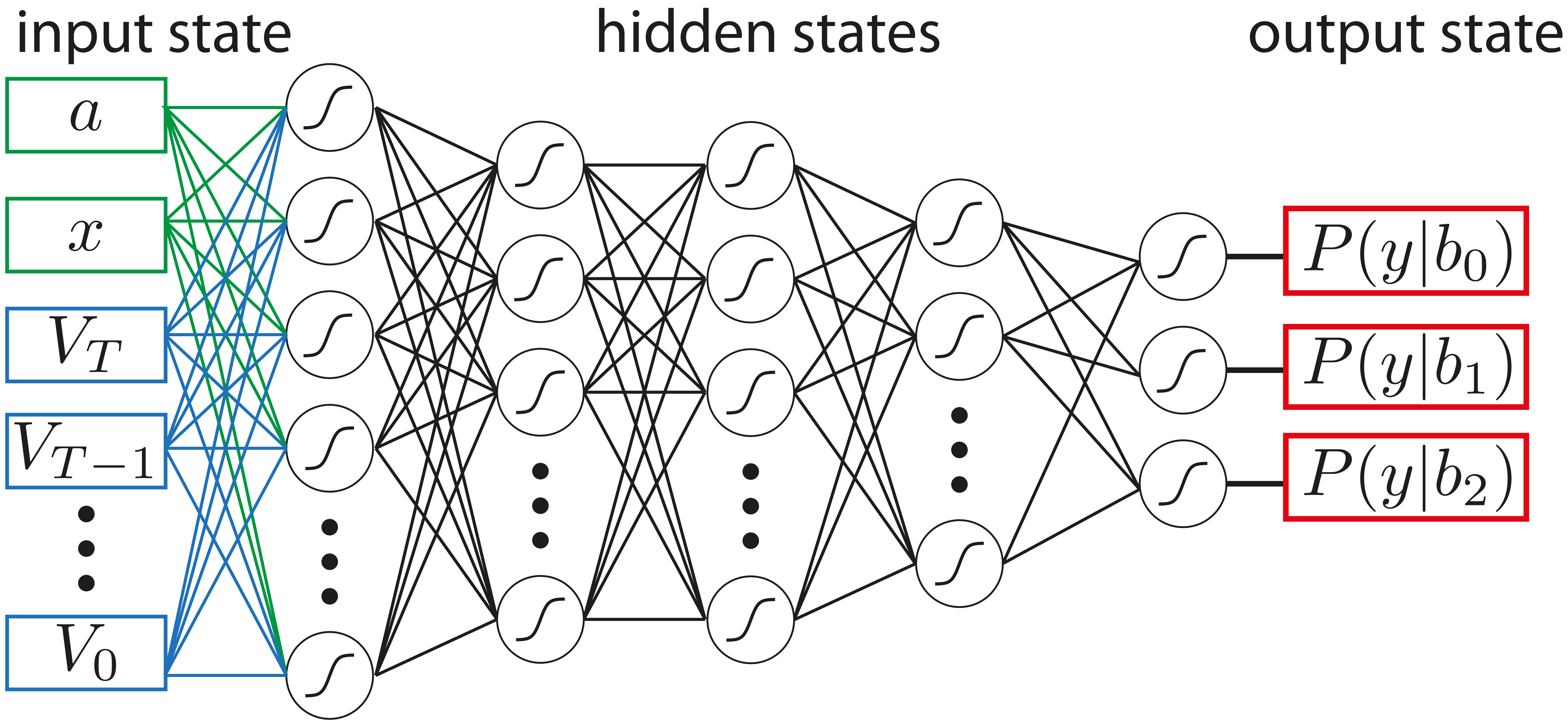
sigmoid activation function

Neural Network are **differentiable computers**

$$\frac{\partial \vec{h}_{n+1}}{\partial \vec{h}_n} = W_n.\partial_x \sigma(W_n.\vec{h}_n + \vec{b})$$

$$\frac{\partial \vec{h}_{n+1}}{\partial W_n} = \vec{h}_n.\partial_x \sigma(W_n.\vec{h}_n + \vec{b})$$

Deep Neural Network



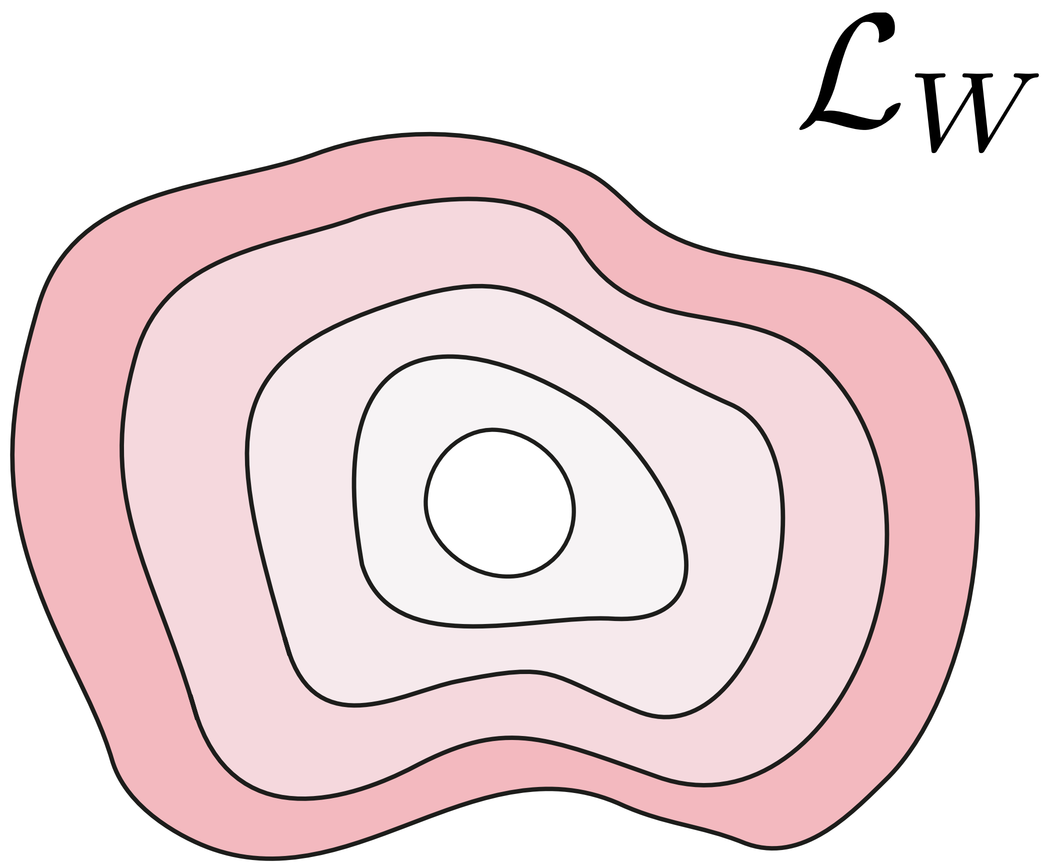
Cross-entropy loss function

$$\mathcal{L} = -y \log P(y|b) - (1-y)\log(1 - P(y|b))$$

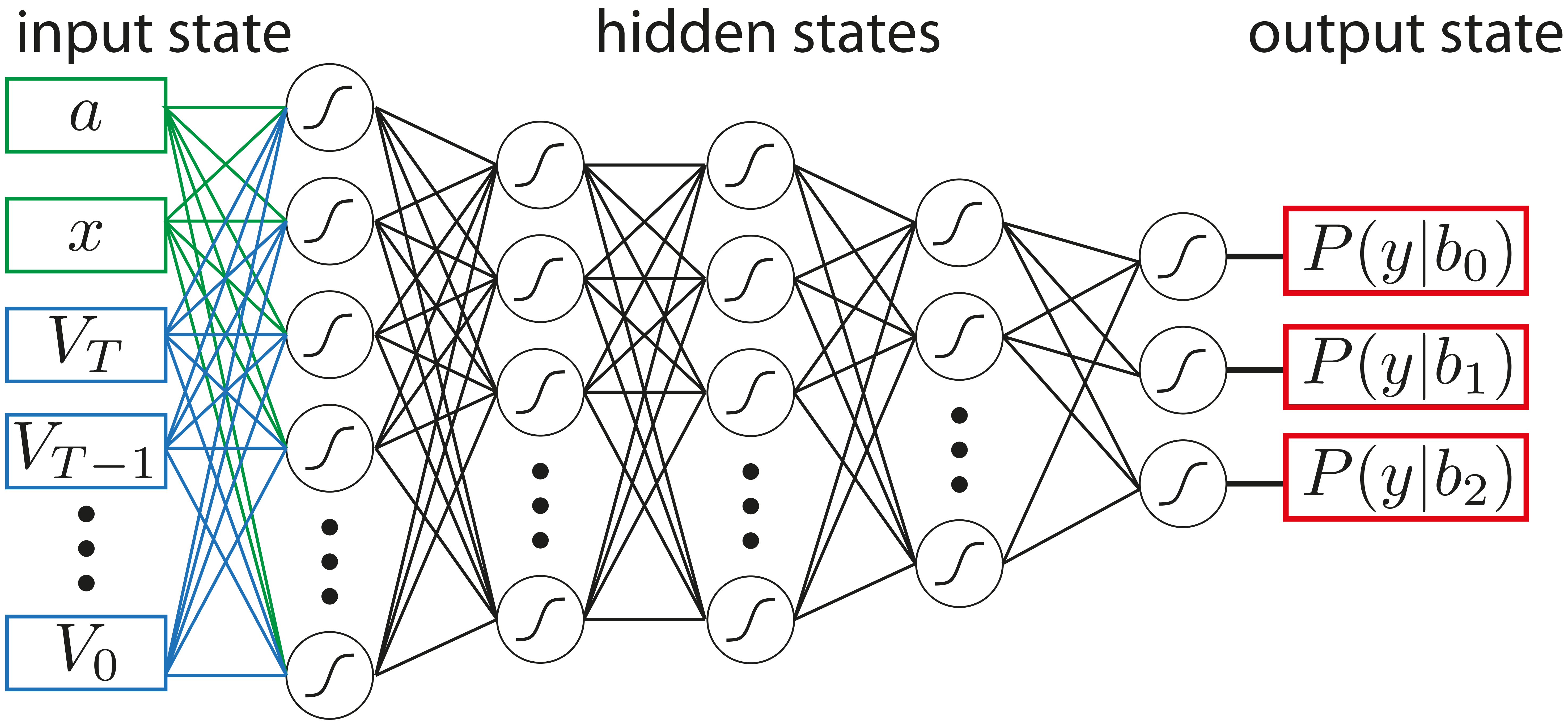
minimum when the distribution of y matches with $P(y)$

measurement bit in the basis b
$y \in \{0, 1\}$

prediction in the basis b



Deep Neural Network

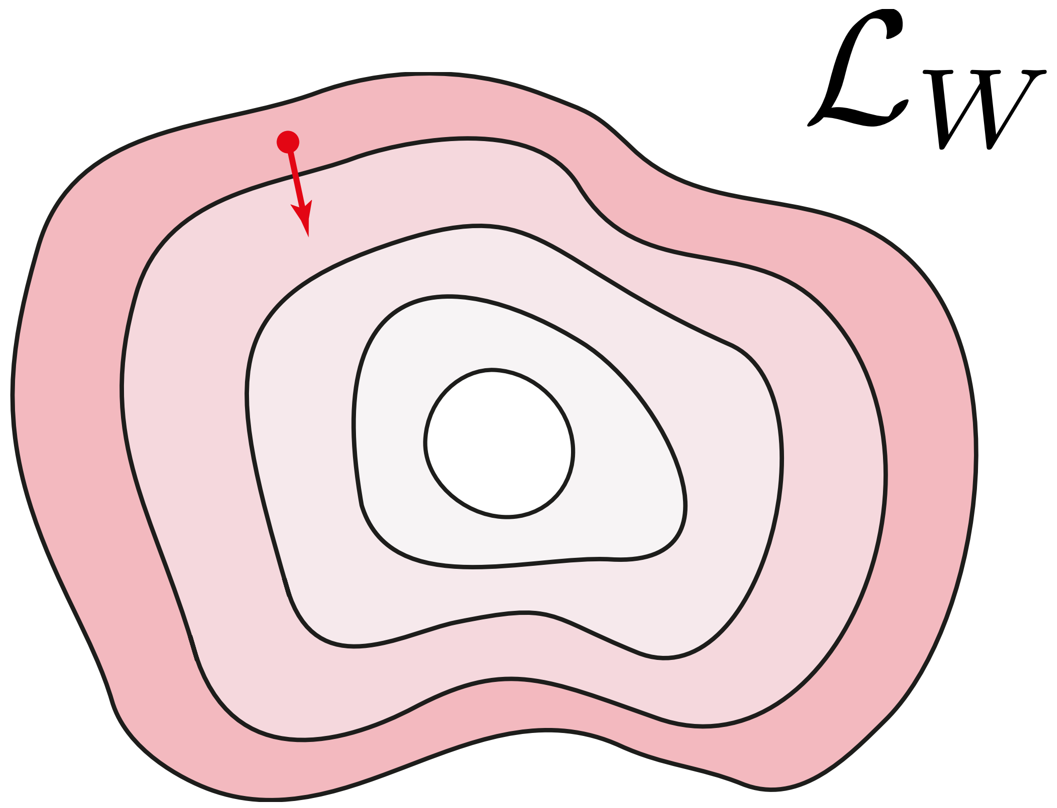


updating all the weight matrices with gradient descent of the loss function

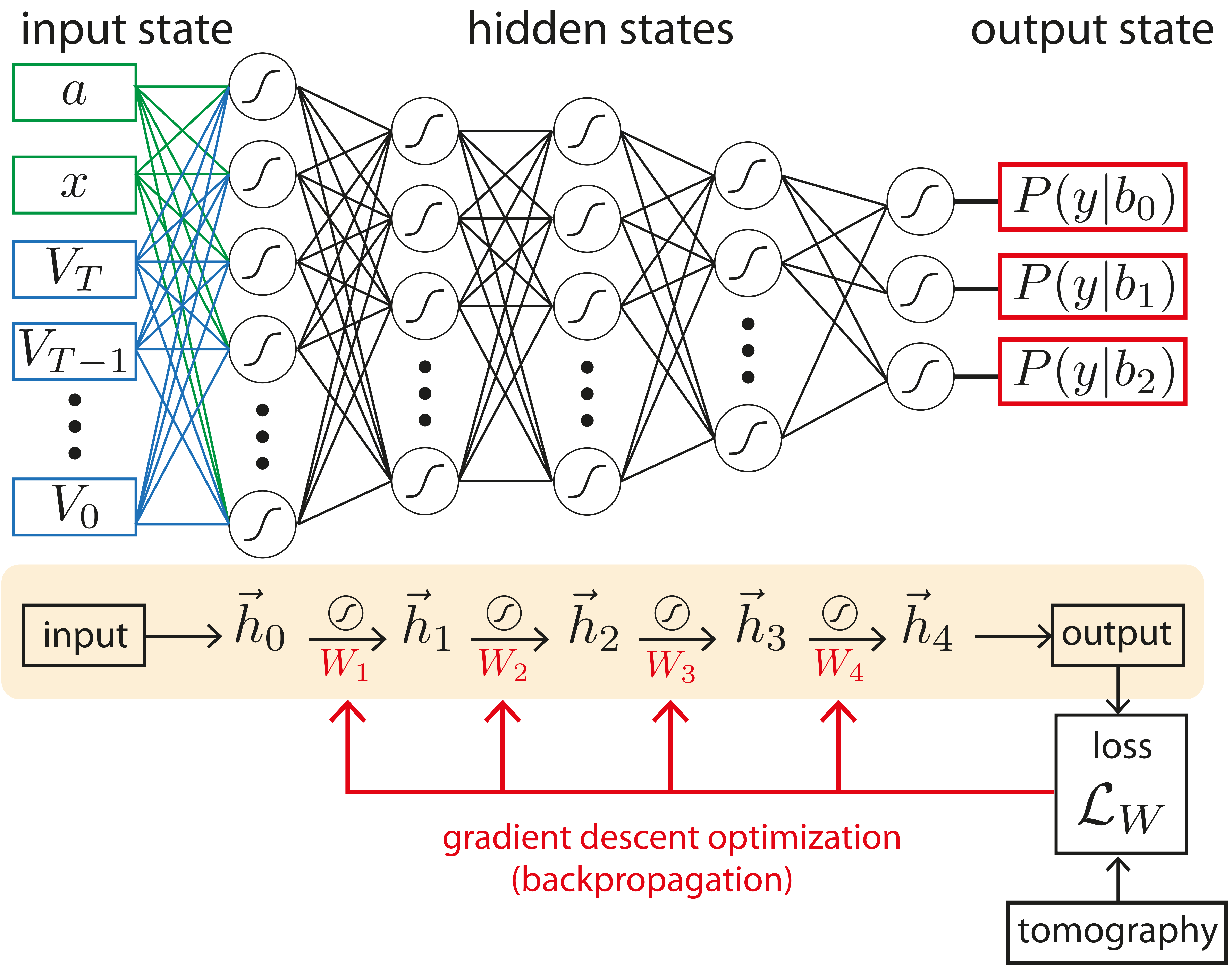
$$W_n \leftarrow W_n - \gamma \frac{\partial \mathcal{L}}{\partial W_n}$$

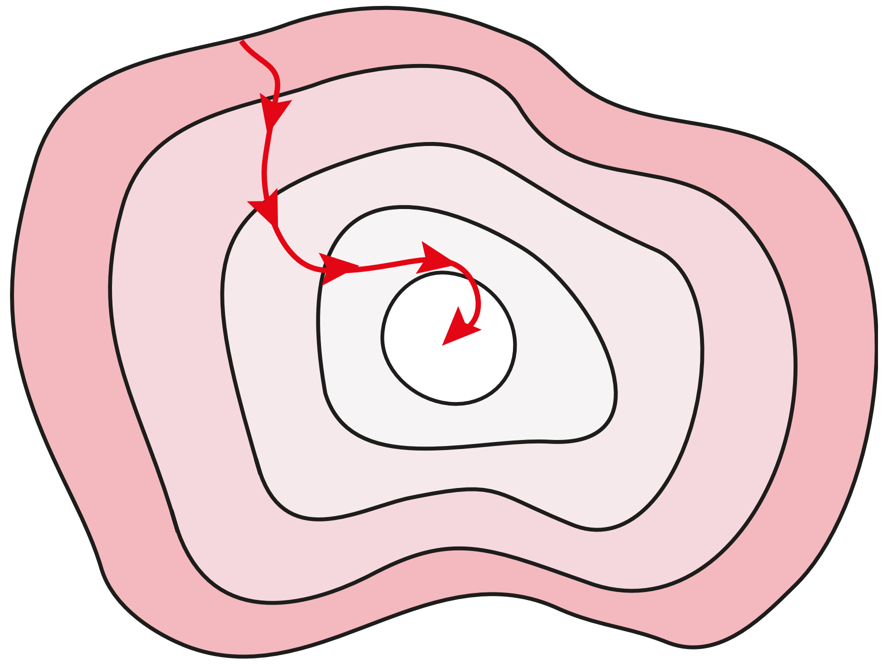
backward propagation $\quad \dfrac{\partial \mathcal{L}}{\partial W_n} = \dfrac{\partial \mathcal{L}}{\partial h_m} \dfrac{\partial h_m}{\partial h_{m-1}} \cdots \dfrac{\partial h_{n+1}}{\partial W_n}$

(the chain rule for differentiation translates in matrix multiplications)



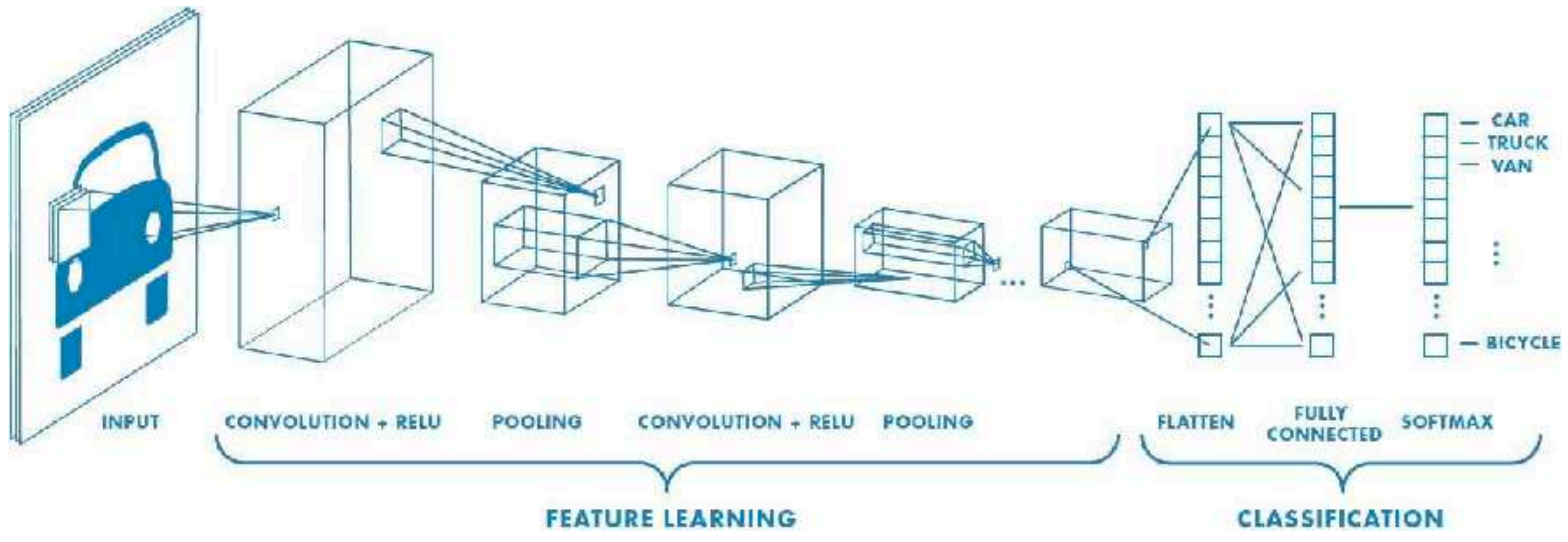
Deep Neural Network





Neural Networks Architectures

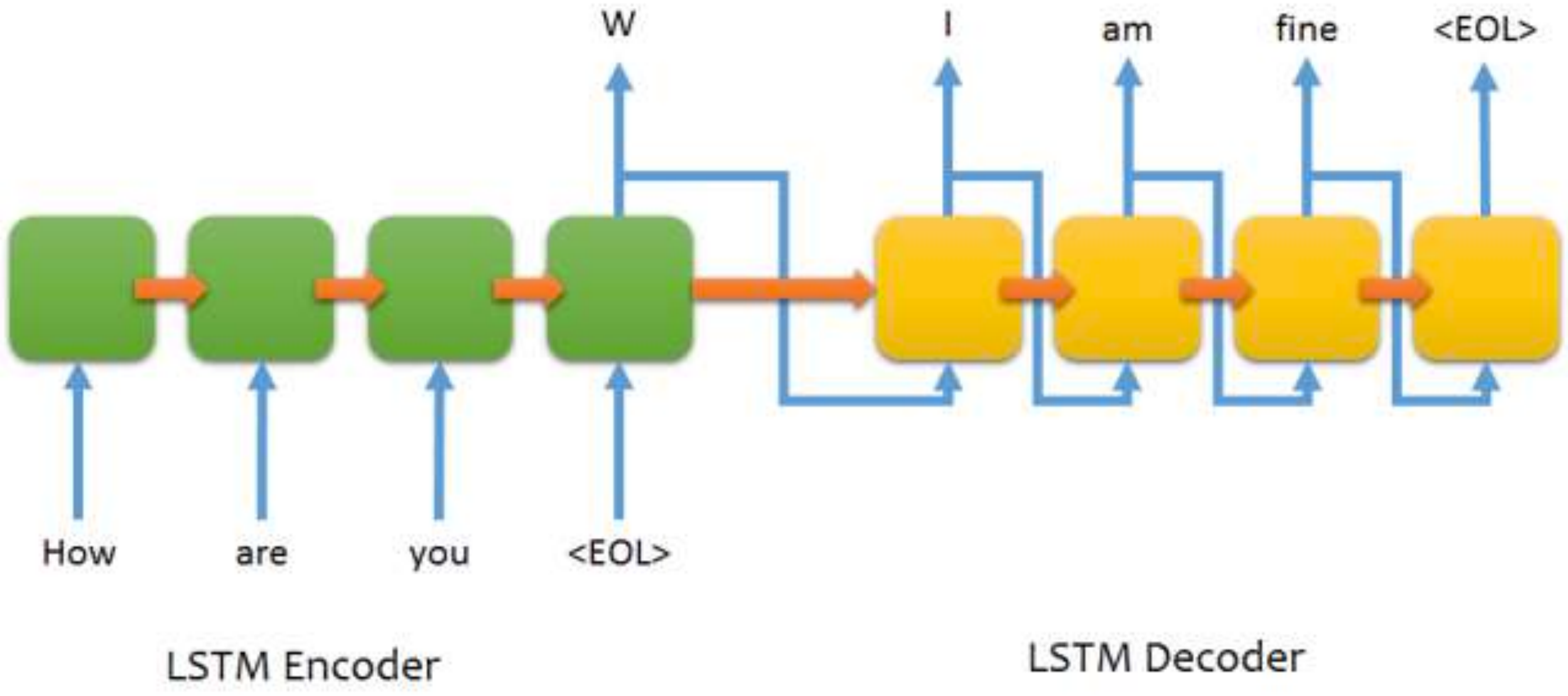
Convolutional Neural Network



Spatially ordered data

- Image recognition

Recurrent Neural Network



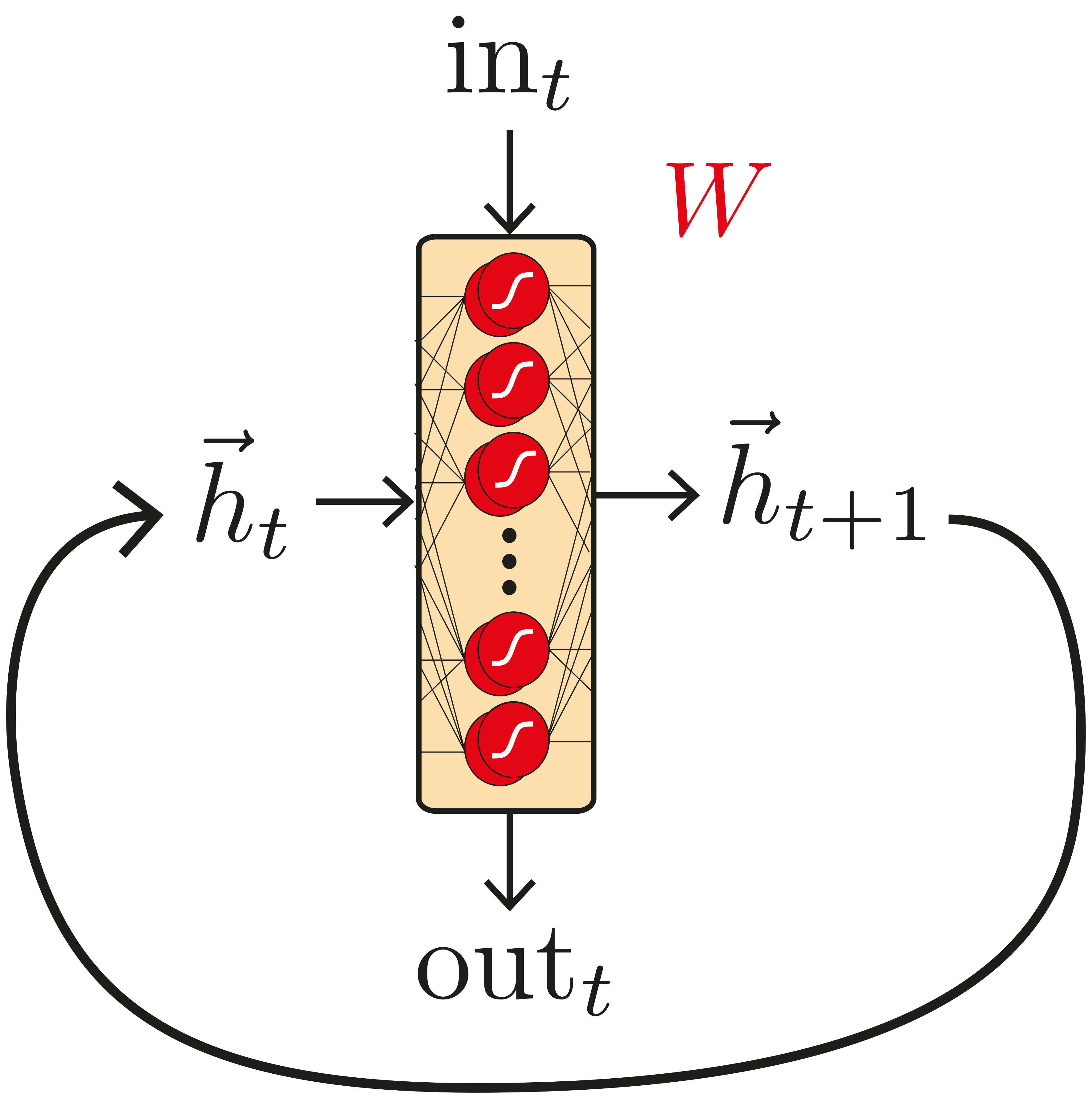
Time series

- text generation
- speech recognition (Amazon Alexa…)
- language translation (Google translate…)

space-like correlations

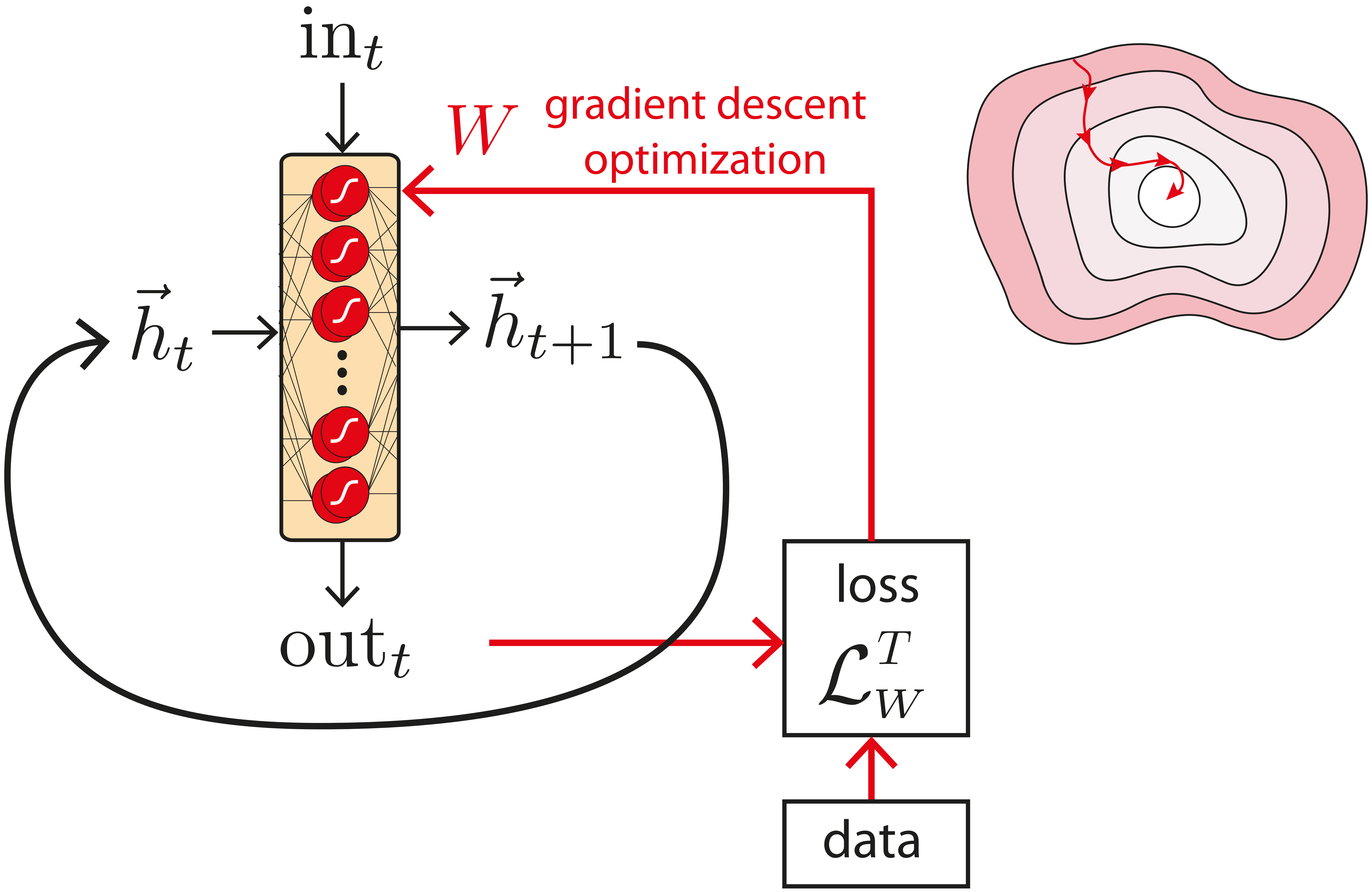
time-like causal correlations

Recurrent Neural Networks

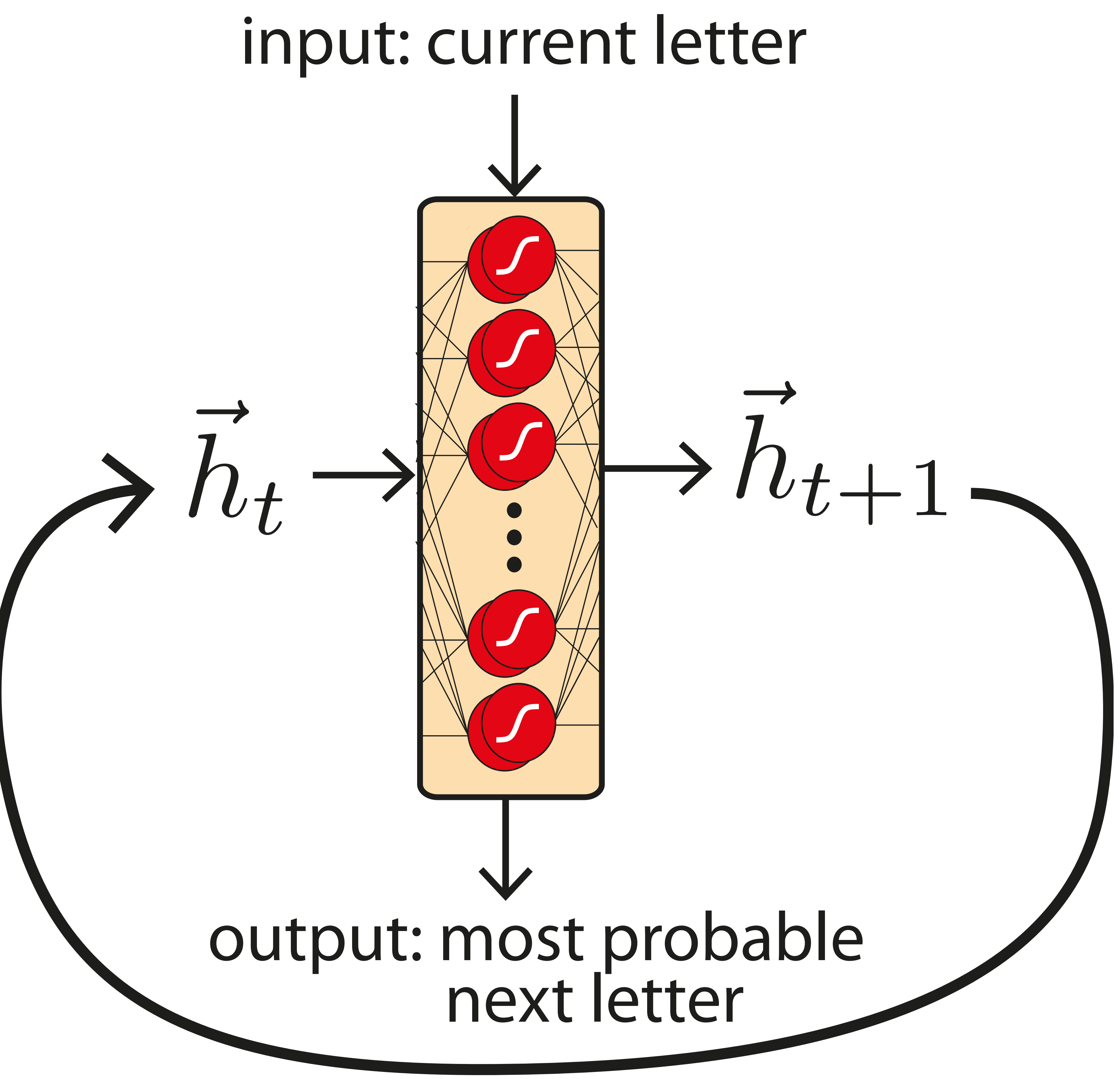


preparation settings inputed in $\vec{h}_0$

Recurrent Neural Networks



Recurrent Neural Networks: Example



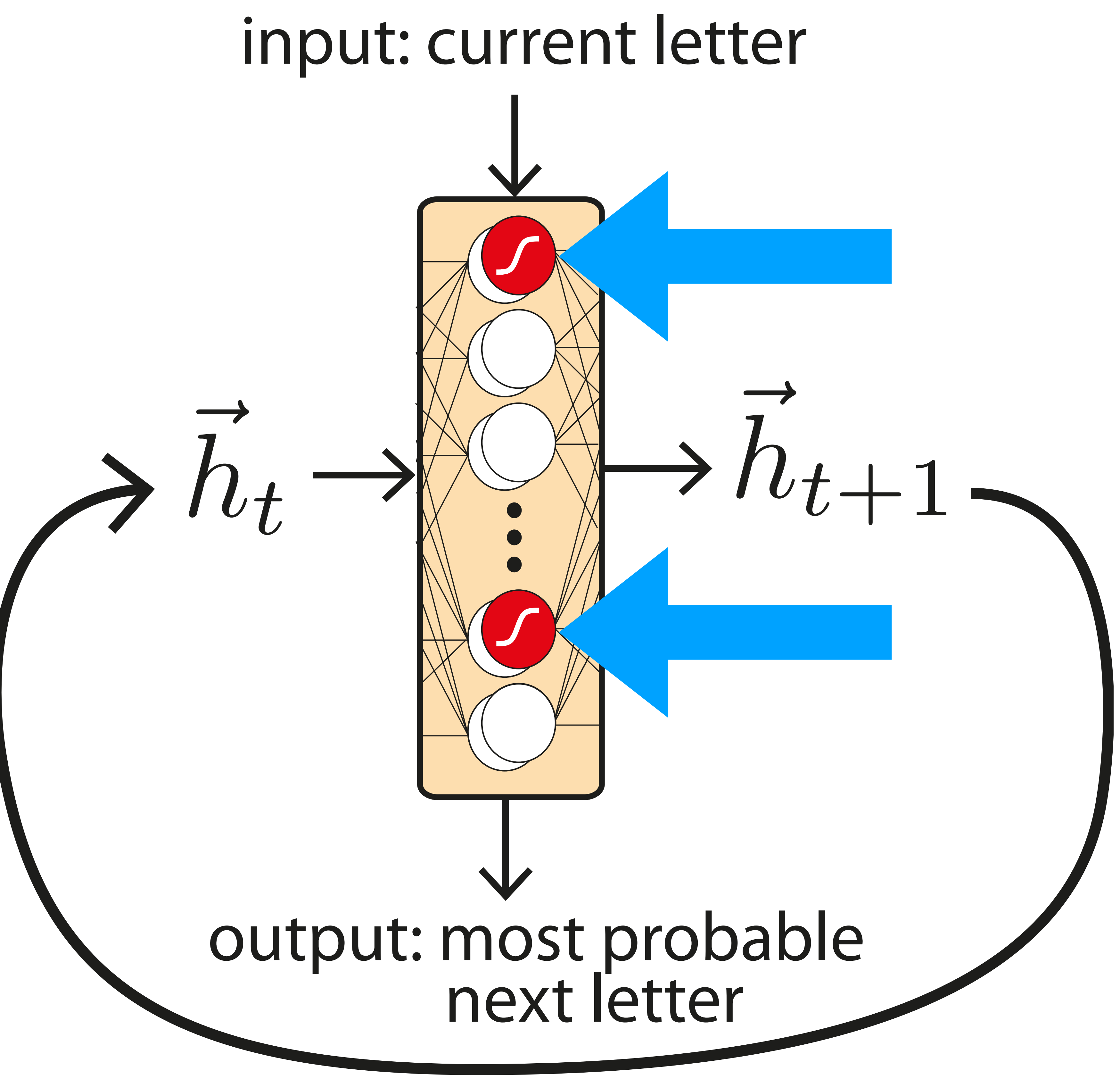
character-level language modeling trained on « War and Peace »

```
The sole importance of the crossing of the Berezina lies in the fact
that it plainly and indubitably proved the fallacy of all the plans for
cutting off the enemy's retreat and the soundness of the only possible
line of action- -the one Kutuzov and the general mass of the army
demanded- -namely, simply to follow the enem?
```

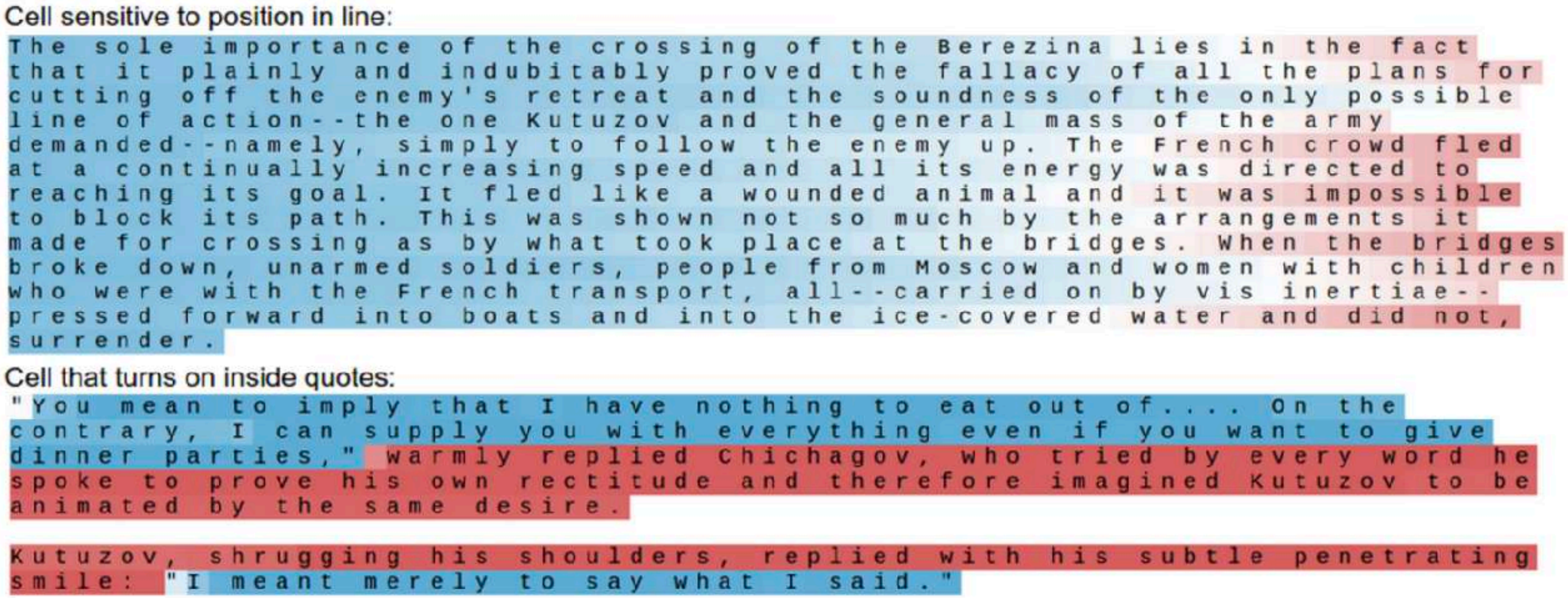
« y »

[Karpathy, Andrej, Justin Johnson, and Li Fei-Fei. "Visualizing and understanding recurrent networks. » *arXiv preprint arXiv:1506.02078* (2015).]

Recurrent Neural Networks: Example



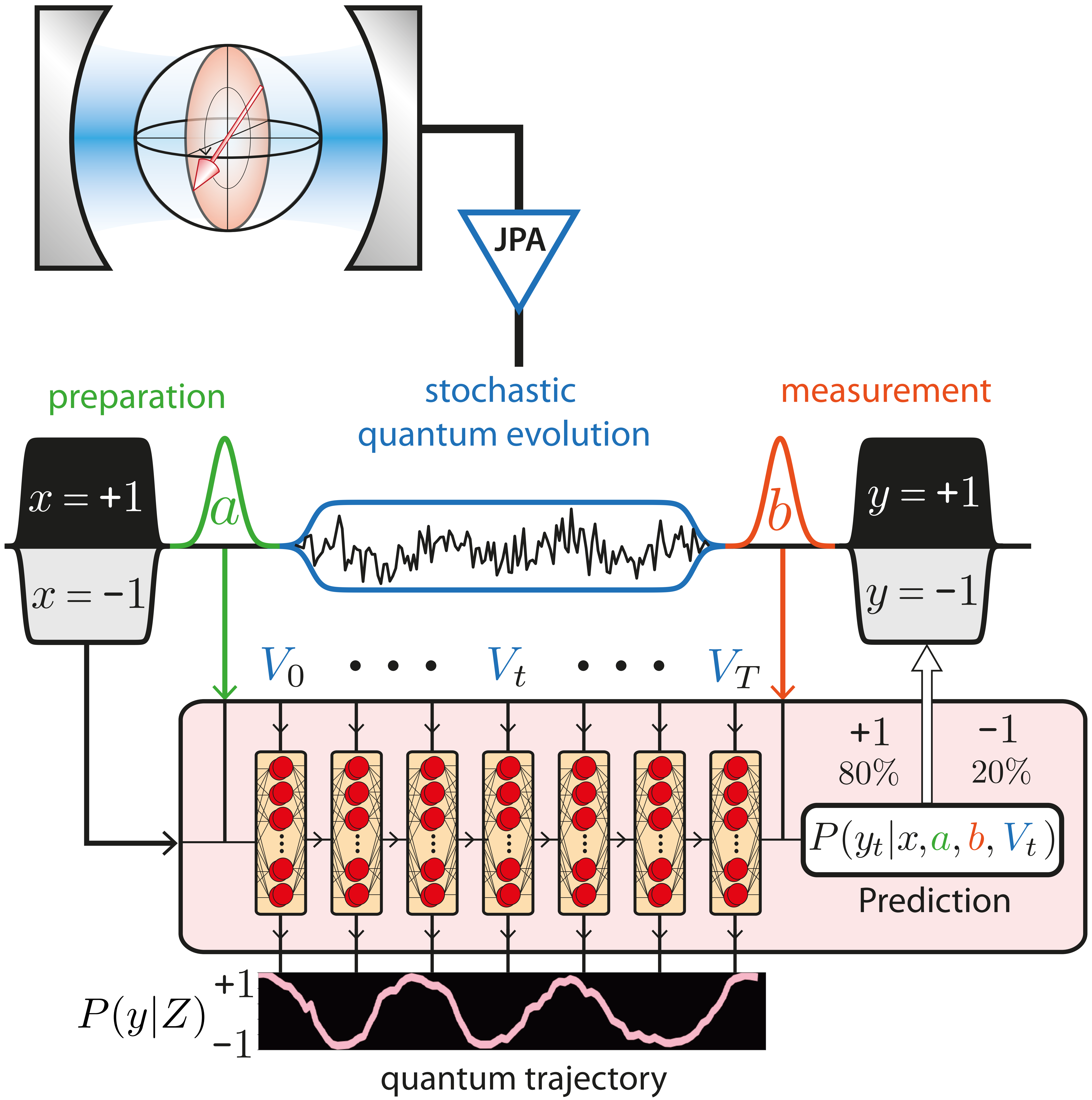
Long range dependencies in RNN



[Karpathy, Andrej, Justin Johnson, and Li Fei-Fei. "Visualizing and understanding recurrent networks.» *arXiv preprint arXiv:1506.02078* (2015).]

RNN

Learning Stochastic Quantum Dynamics



Training set
6 preparation settings
6 measurement settings
20 experiment durations

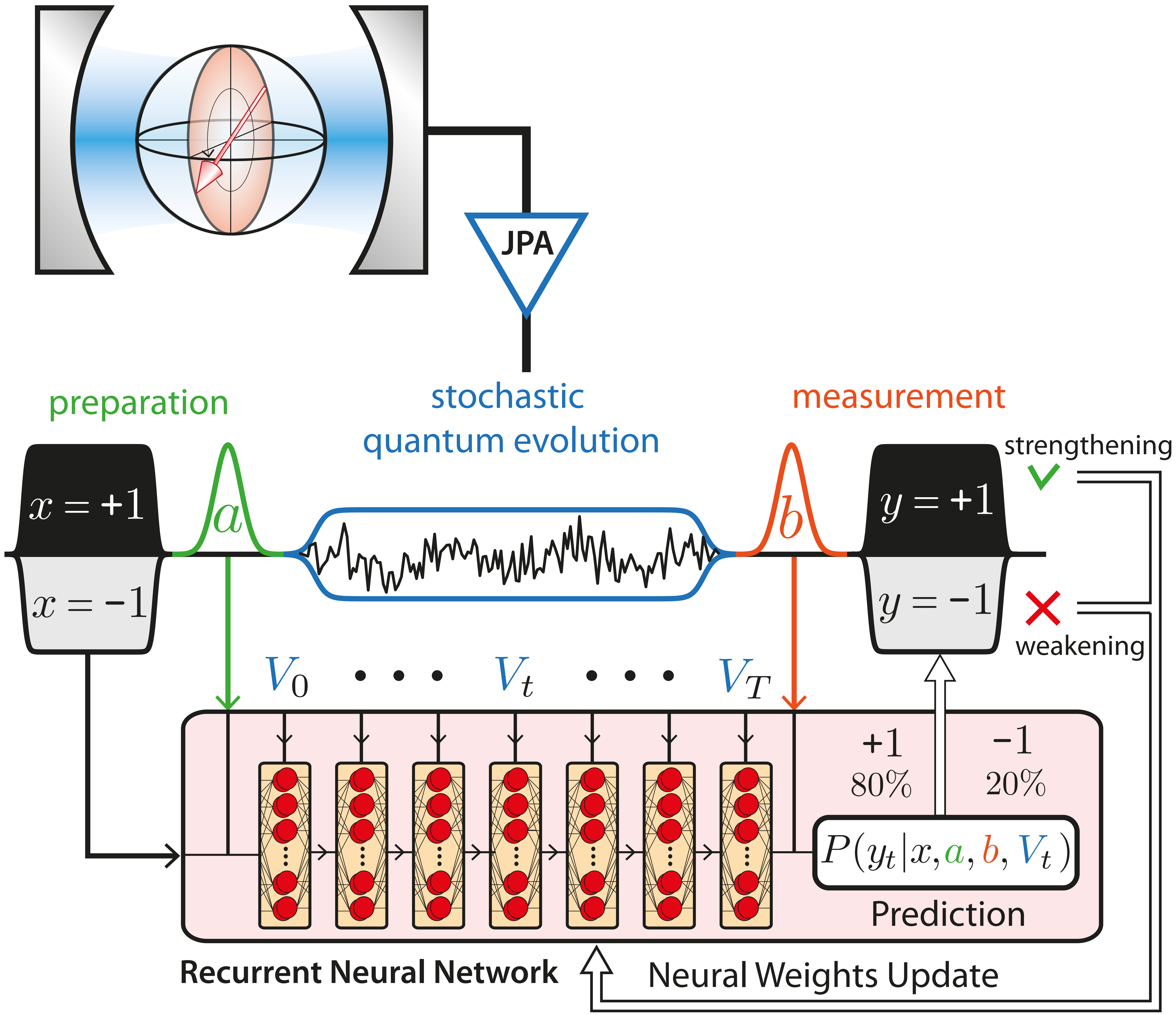
Experiment
1.5 millions repetitions
at a rate of **0.5 ms**

Recurent Neural Network
Long Short Term Memory
32 neurones per layer
5,000 weights parameters
0.8 ms of GPU training per trace

LSTM babysitting
batchsize 1024
10 epochs
learning rate 10^{-3}->10^{-6}
dropout 0.3->0

RNN

Learning Stochastic Quantum Dynamics



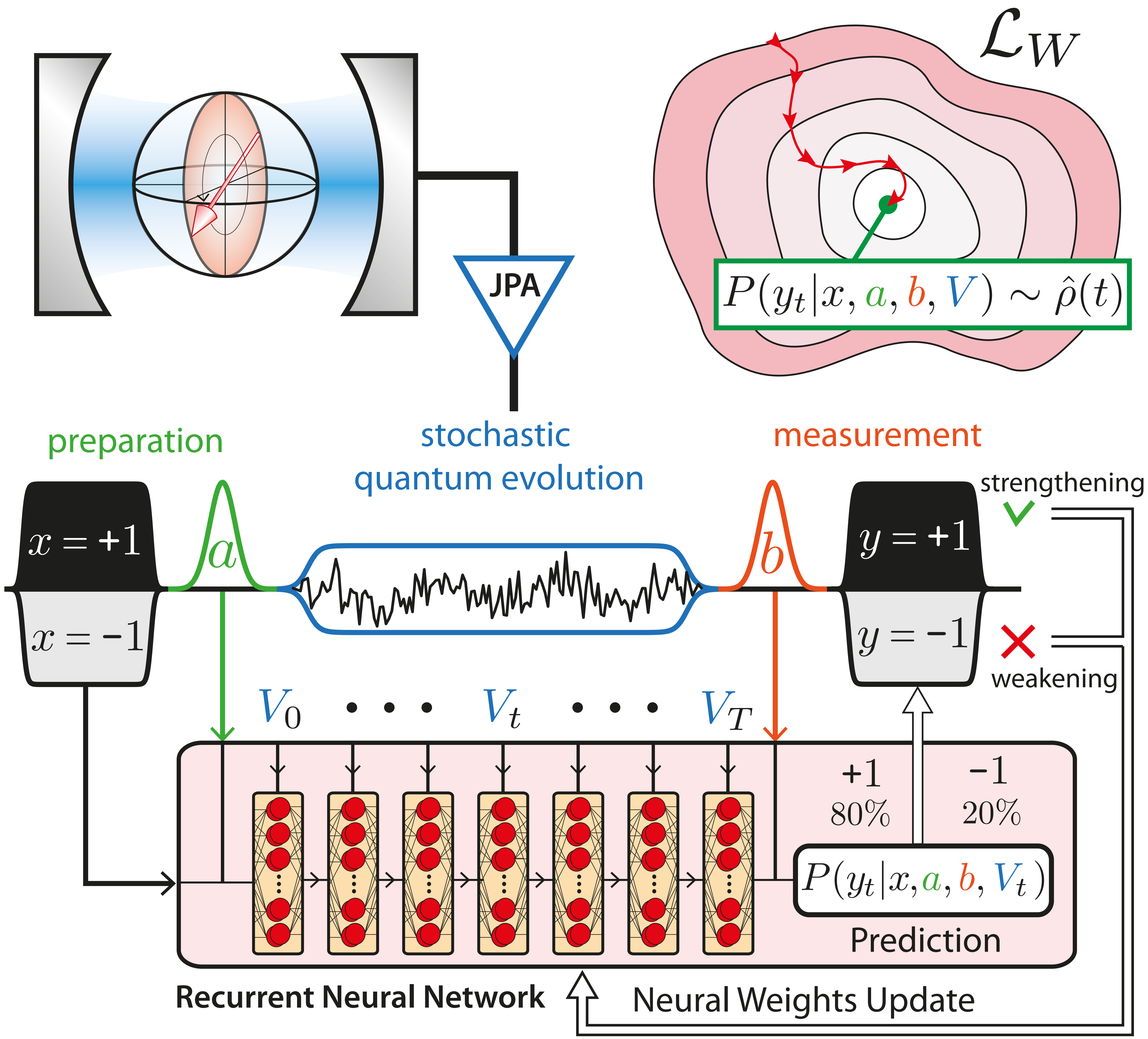
Experiment
1.5 millions repetitions
at a rate of **0.5 ms**

Recurent Neural Network
Long Short Term Memory
32 neurones per layer
5,000 weights parameters
0.8 ms of GPU training per trace

Training set
6 preparation settings
6 measurement settings
20 experiment durations

RNN

Learning Stochastic Quantum Dynamics



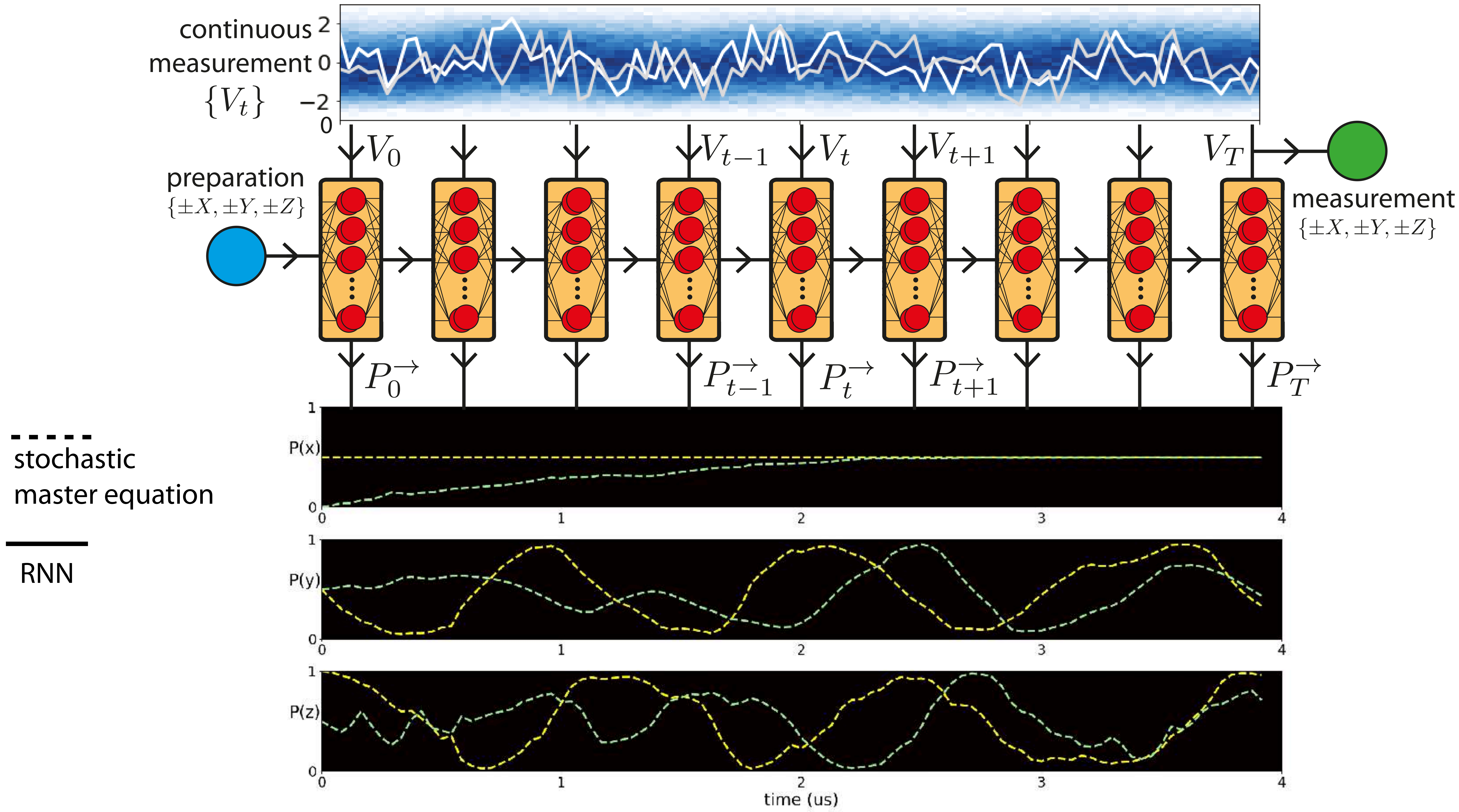
Experiment
1.5 millions repetitions
at a rate of **0.5 ms**

Recurent Neural Network
Long Short Term Memory
32 neurones per layer
5,000 weights parameters
0.8 ms of GPU training per trace

Training set
6 preparation settings
6 measurement settings
20 experiment durations

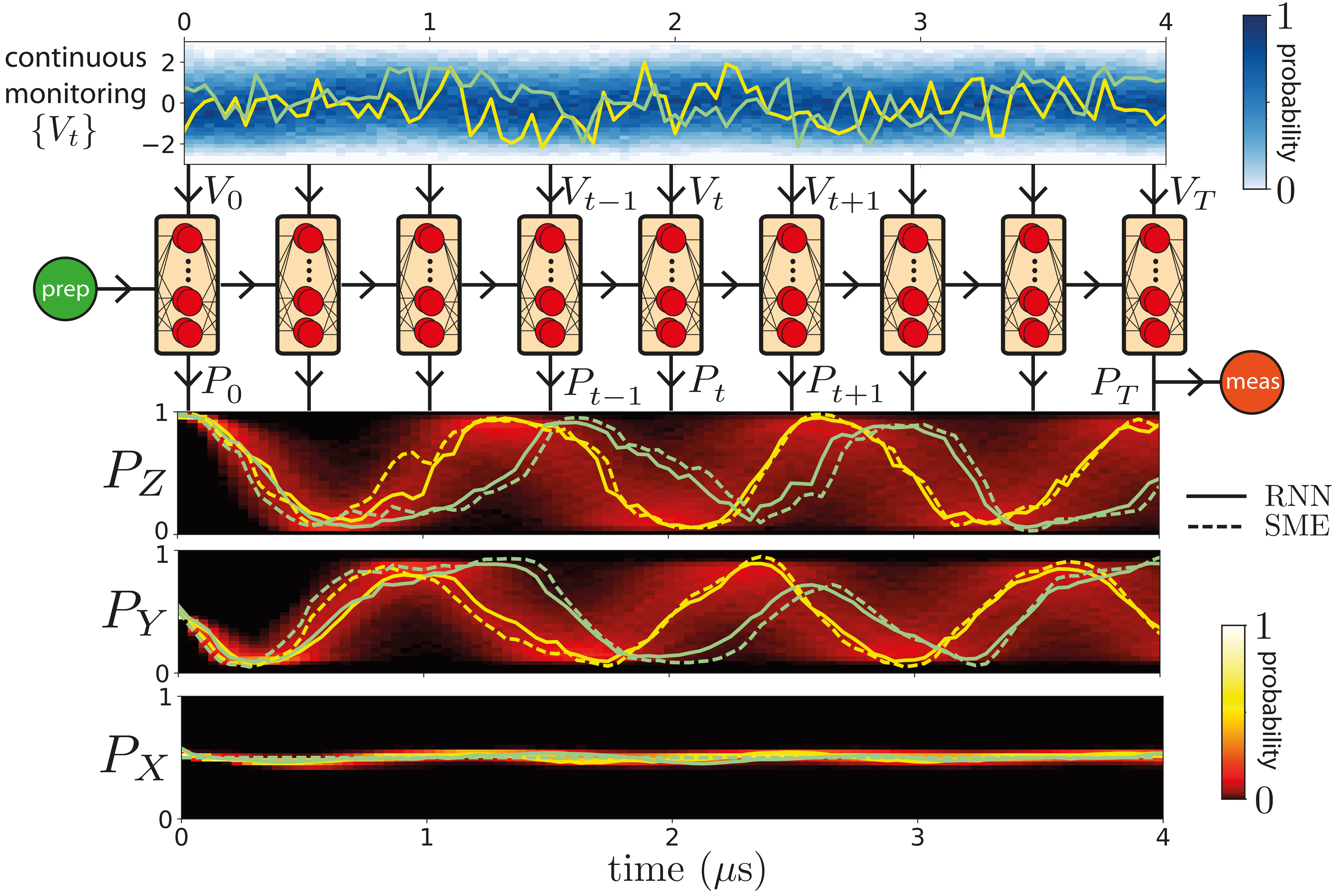
RNN

Training



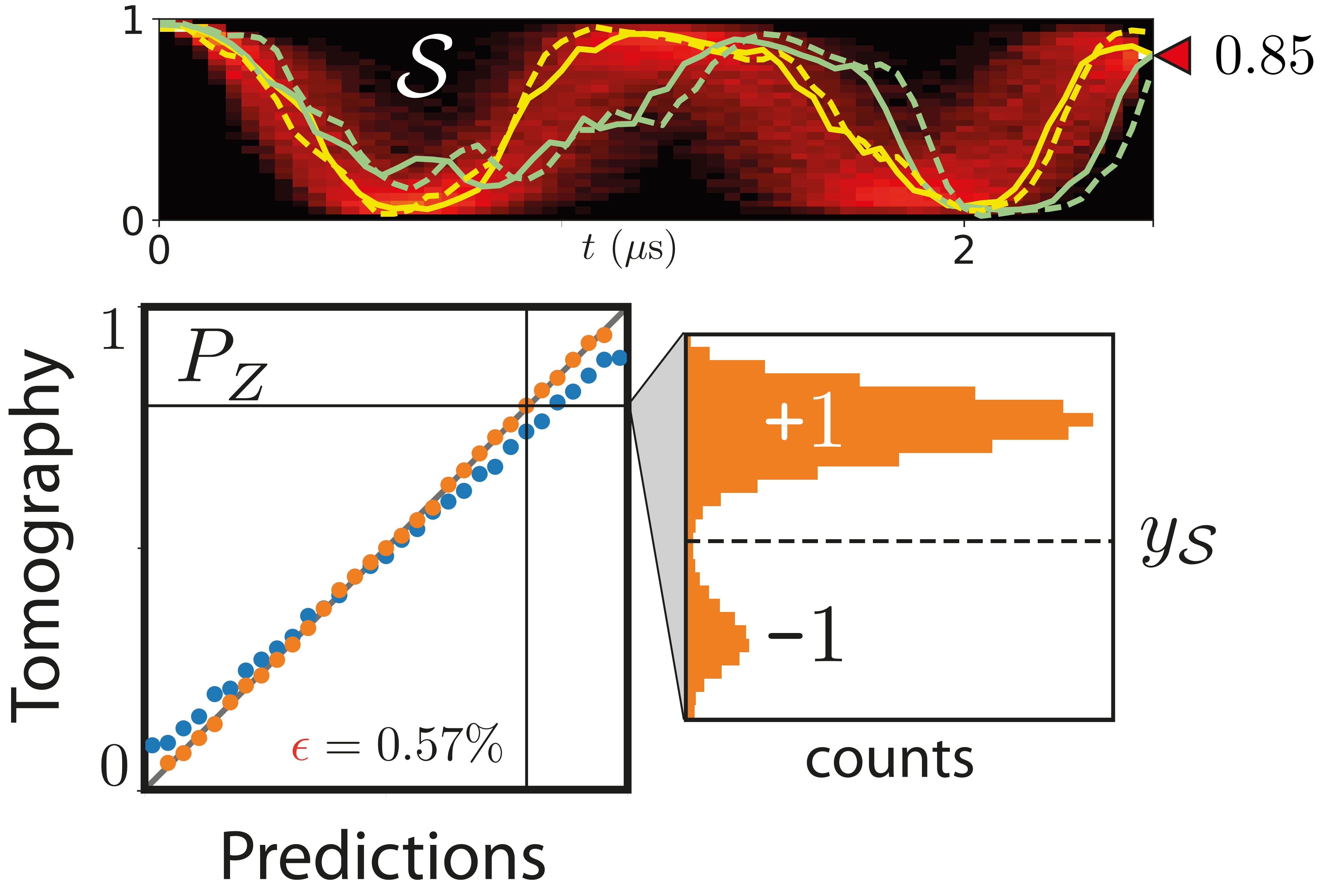
RNN

Learning Quantum Mechanics



Training Validation

training validation

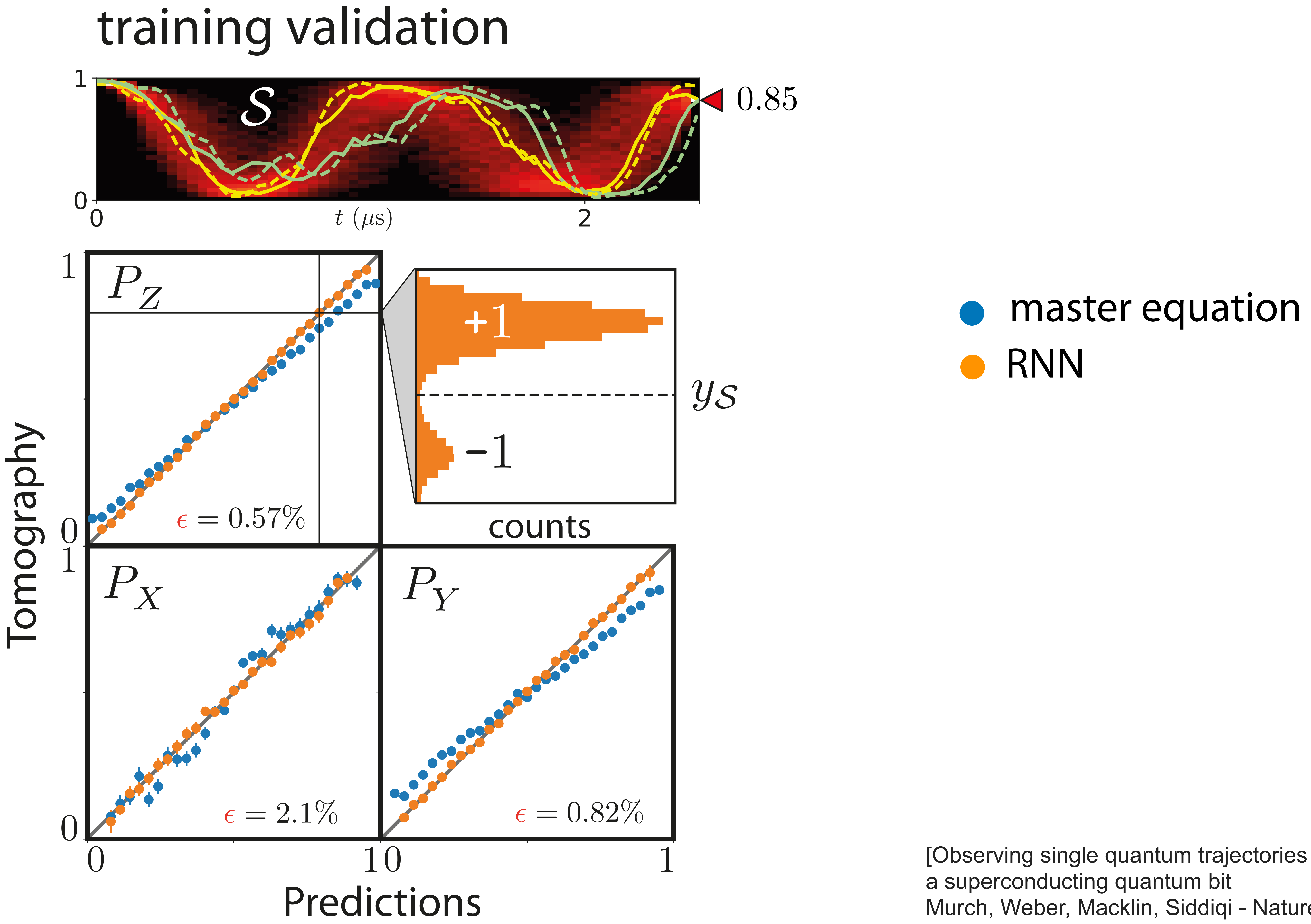


master equation

RNN

[Observing single quantum trajectories of
a superconducting quantum bit
Murch, Weber, Macklin, Siddiqi - Nature (2013)]

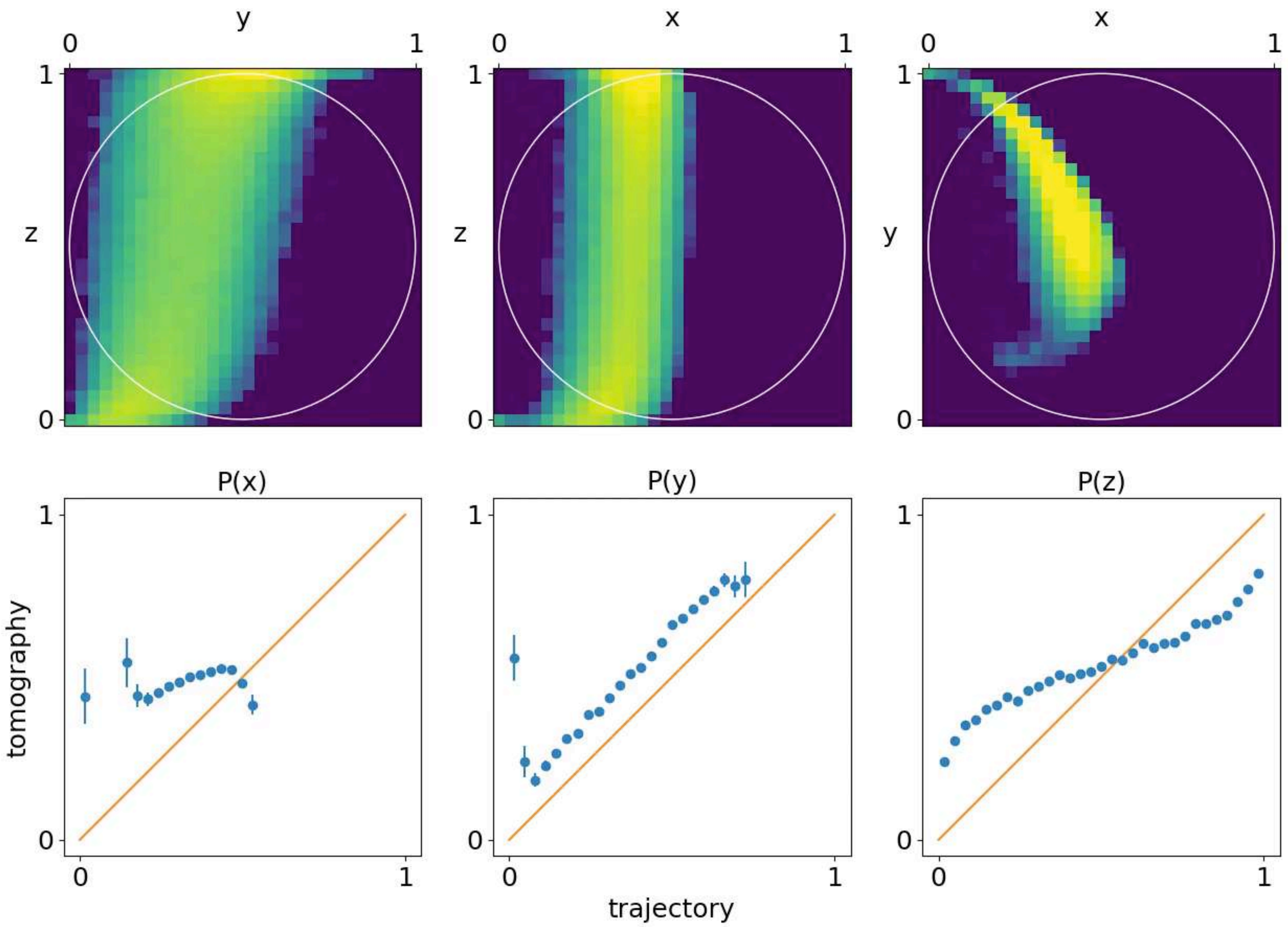
Training Validation



[Observing single quantum trajectories of a superconducting quantum bit
Murch, Weber, Macklin, Siddiqi - Nature (2013)]

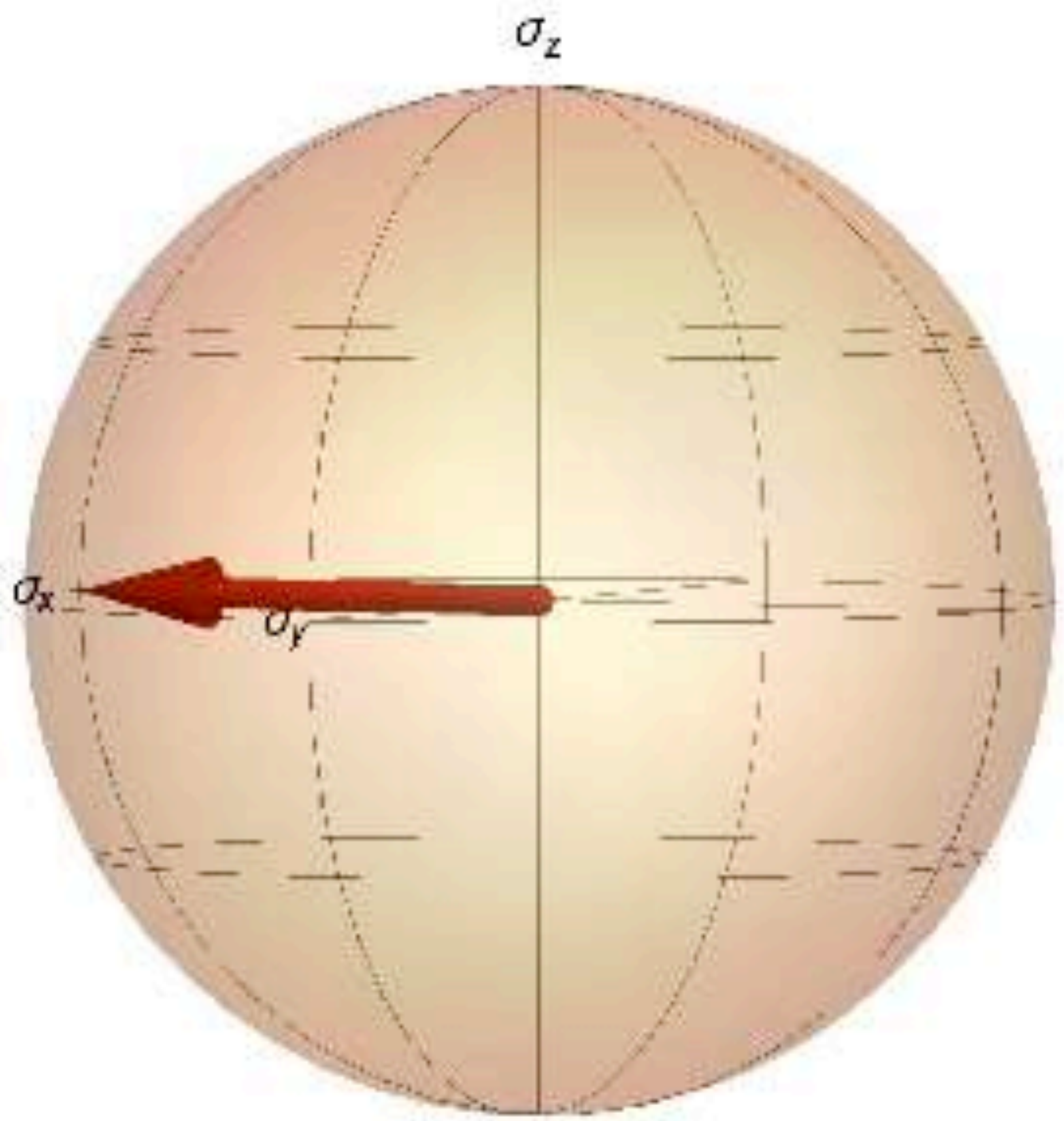
RNN

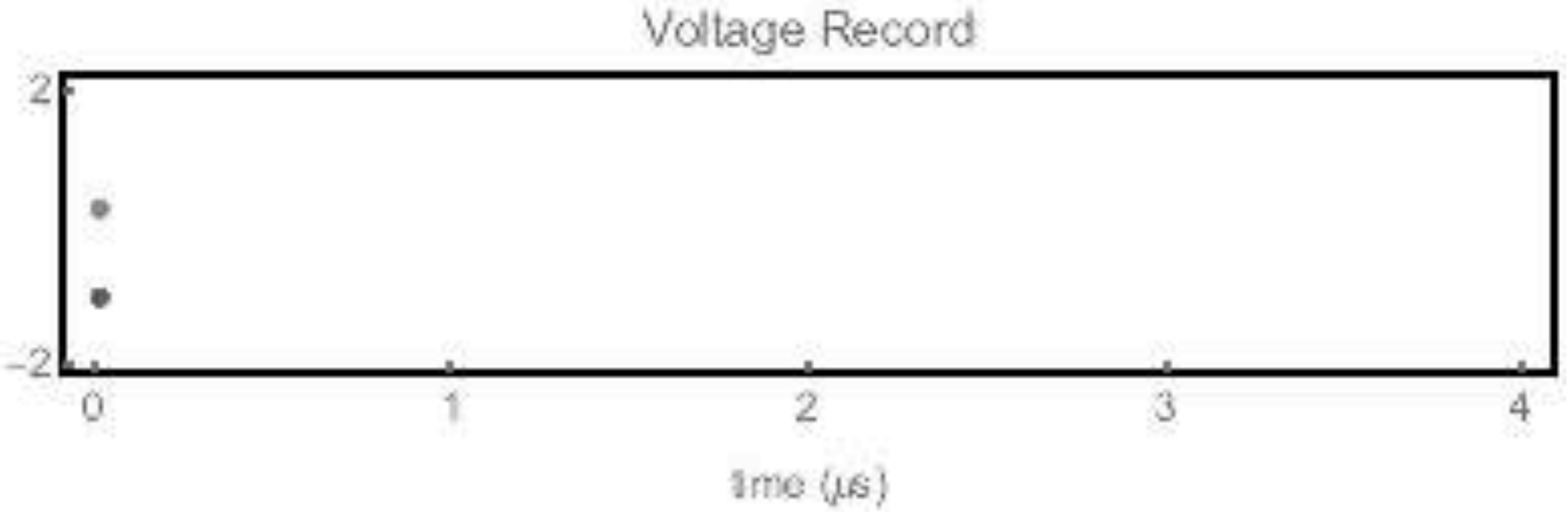
Training Validation

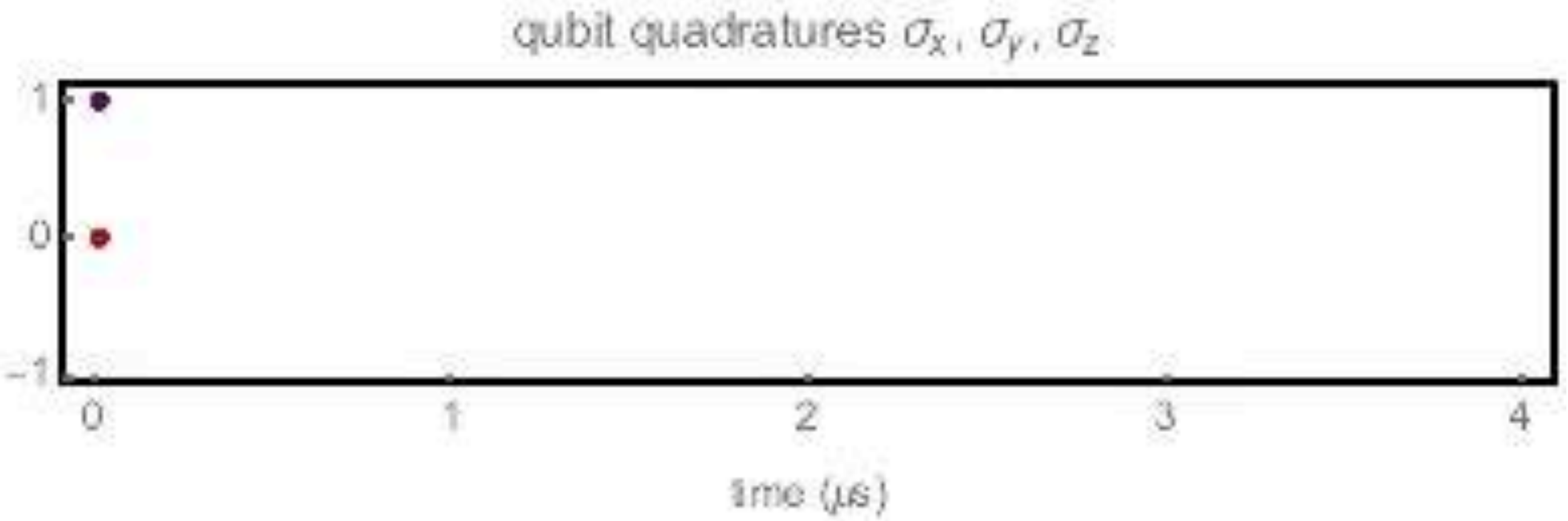


RNN

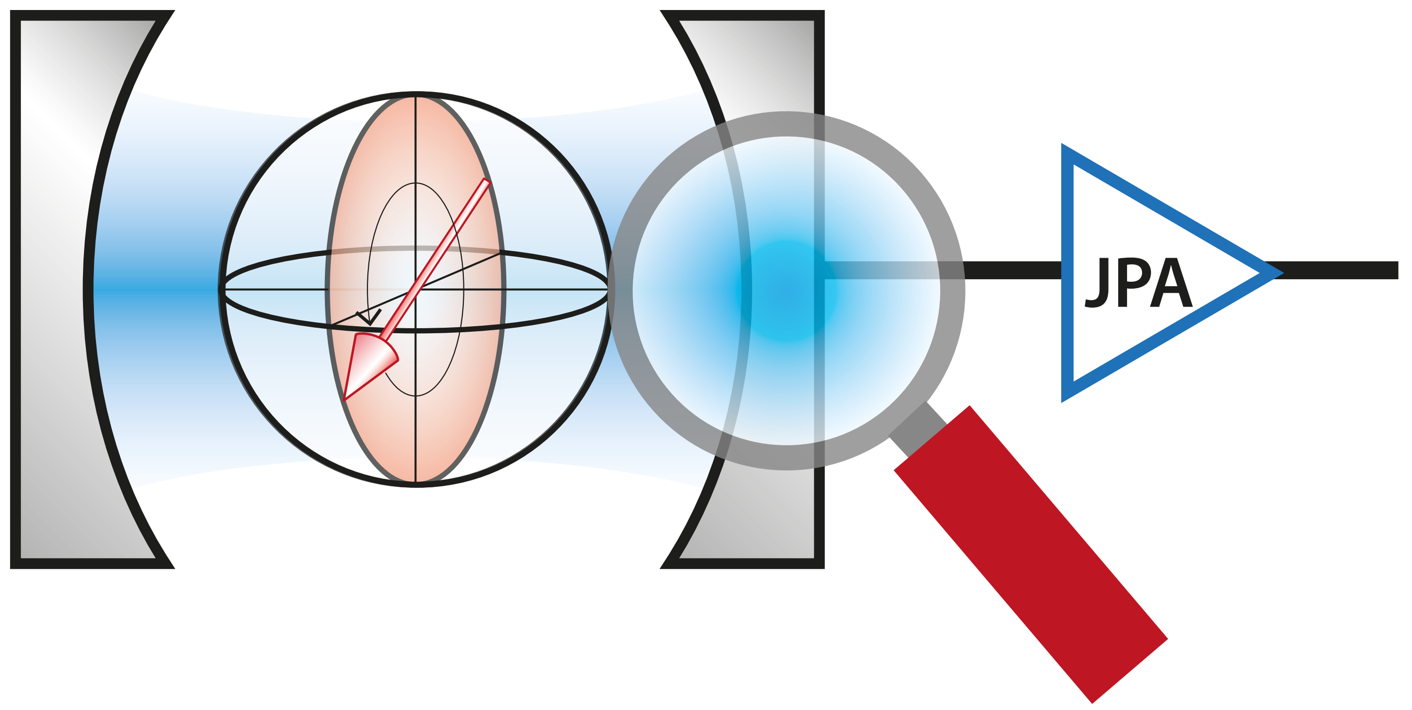
Machine Prediction

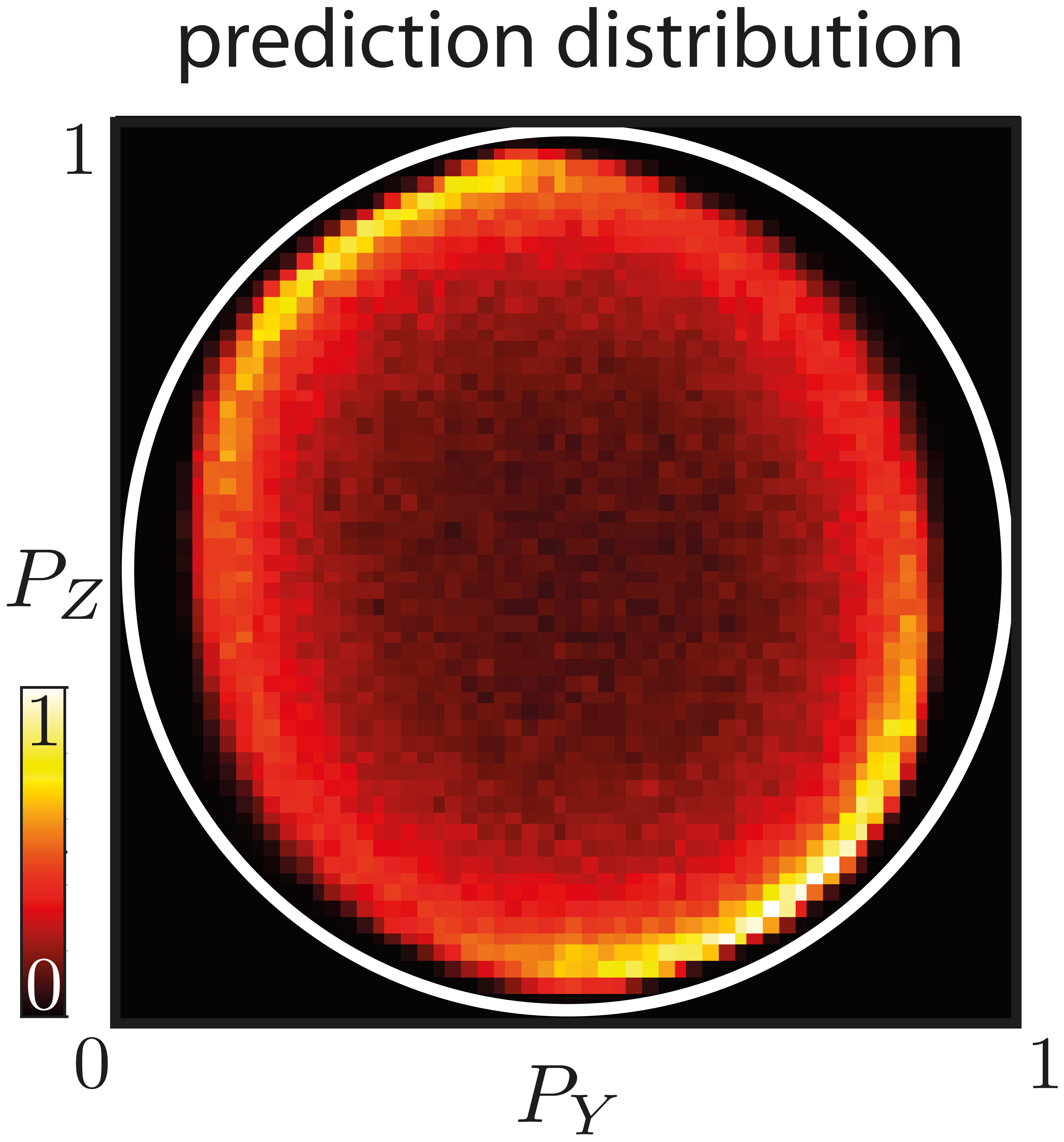




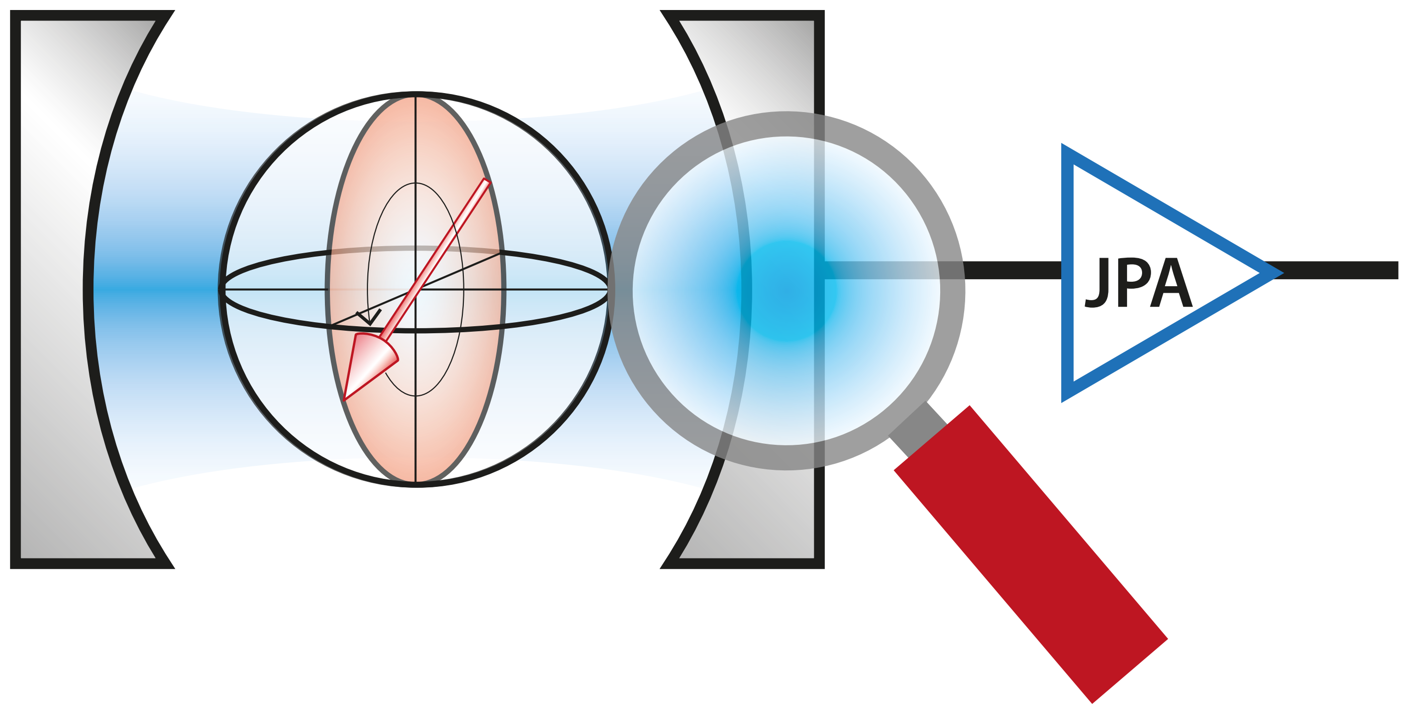


Parameter Estimation & Sensing



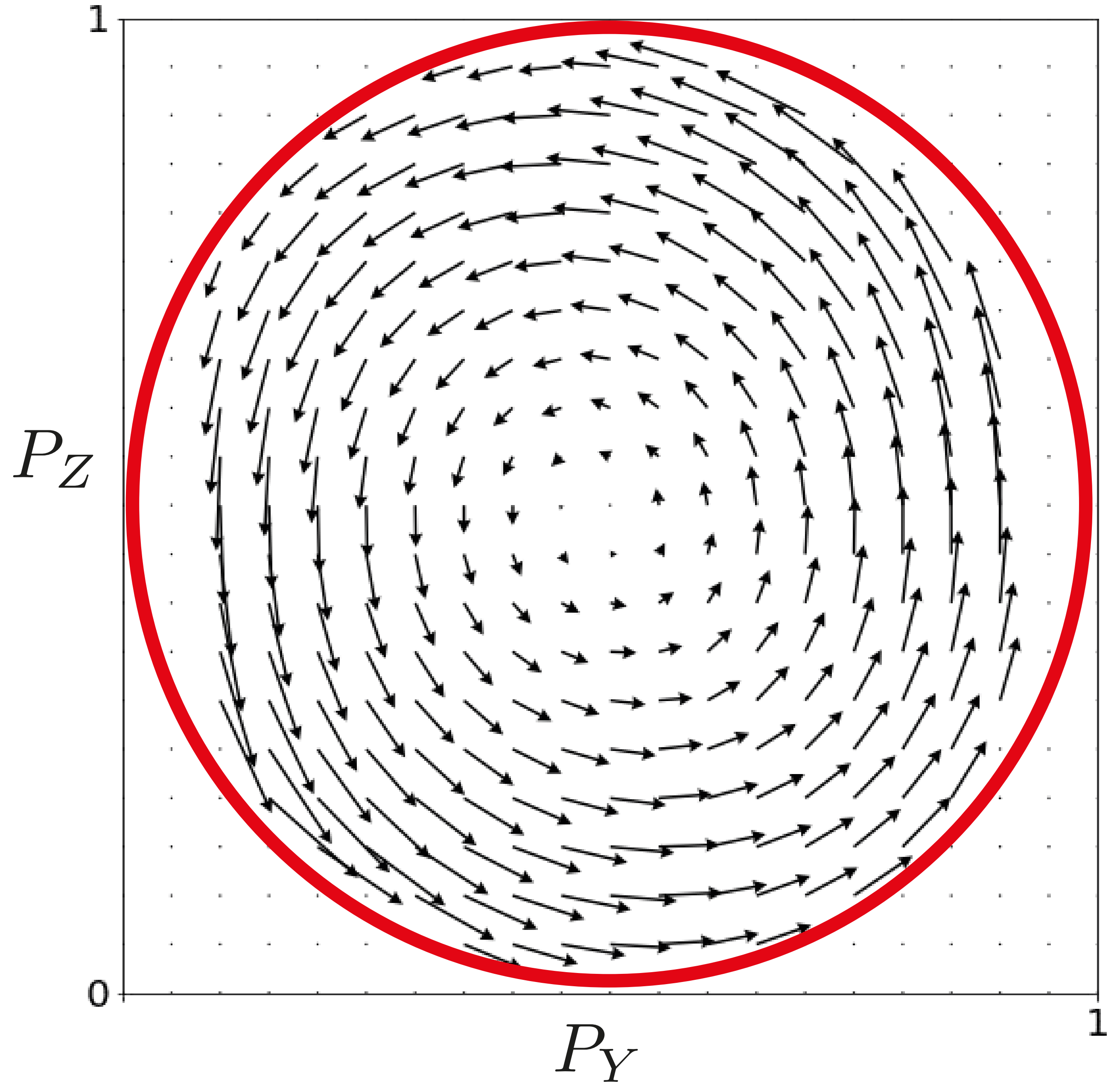


Parameter Estimation & Sensing



Drift map

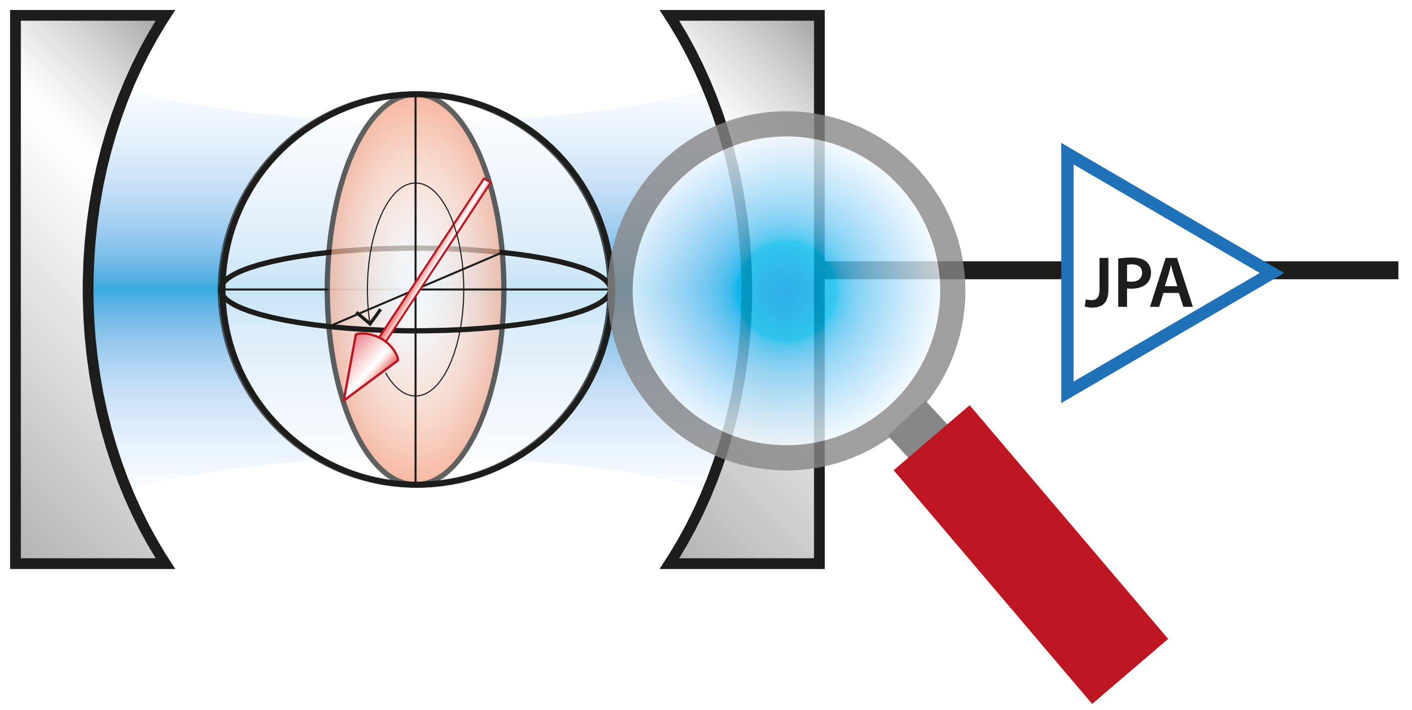
$$\langle \vec{P}_{t+dt} - \vec{P}_t \rangle$$



Dissipative evolution

$$\partial_t \rho = \left(i[\rho, H] + \sum_\sigma [\sigma^\dagger \rho \sigma - \frac{1}{2}(\sigma^\dagger \sigma \rho + \rho \sigma^\dagger \sigma)] \right) \mathbf{dt}$$

Parameter Estimation & Sensing

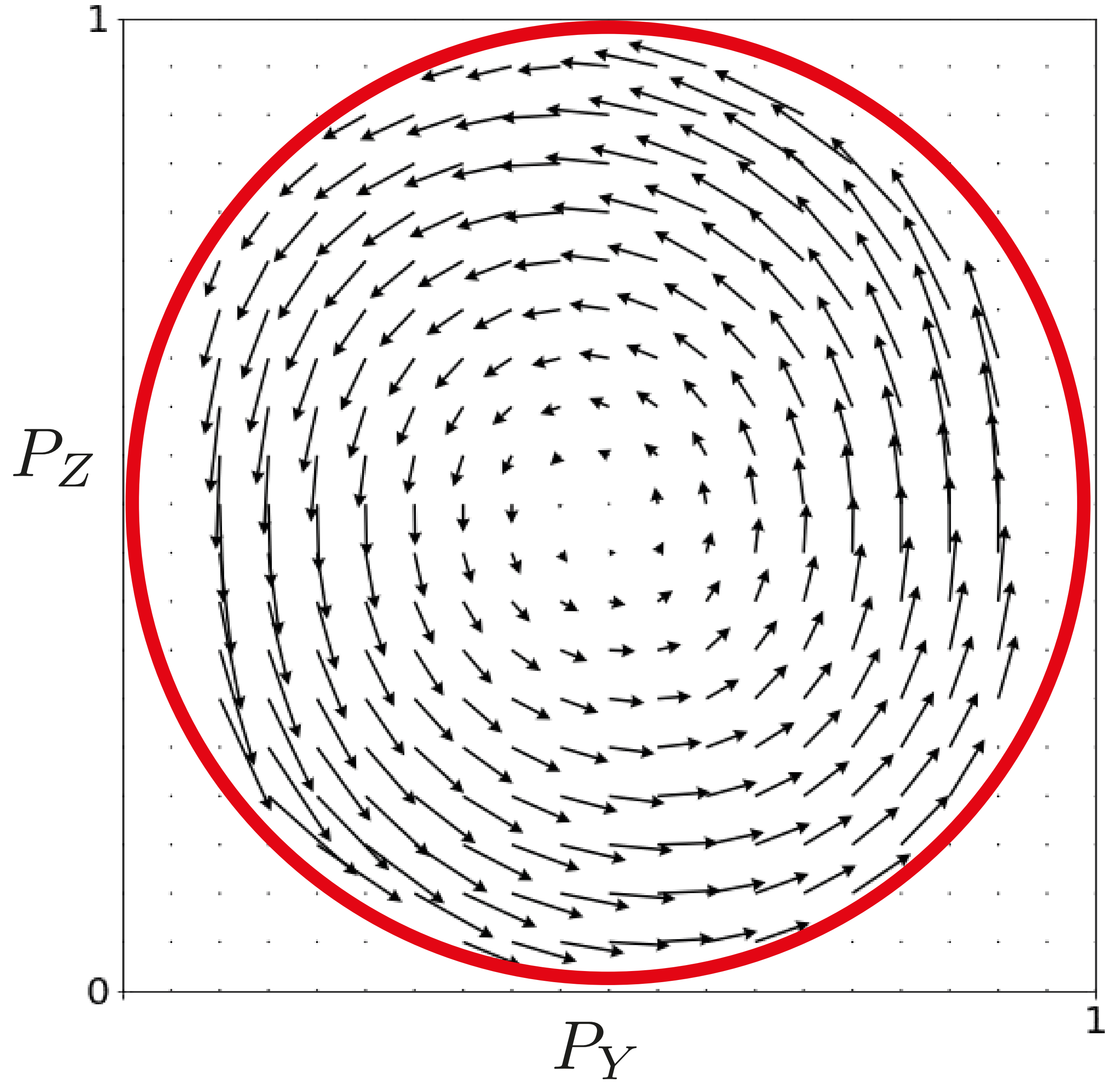


Dephasing rate
$\gamma = 0.82$ MHz

Rabi frequency
$\Omega_R = 1.10$ MHz

Drift map

$\langle \vec{P}_{t+dt} - \vec{P}_t \rangle$

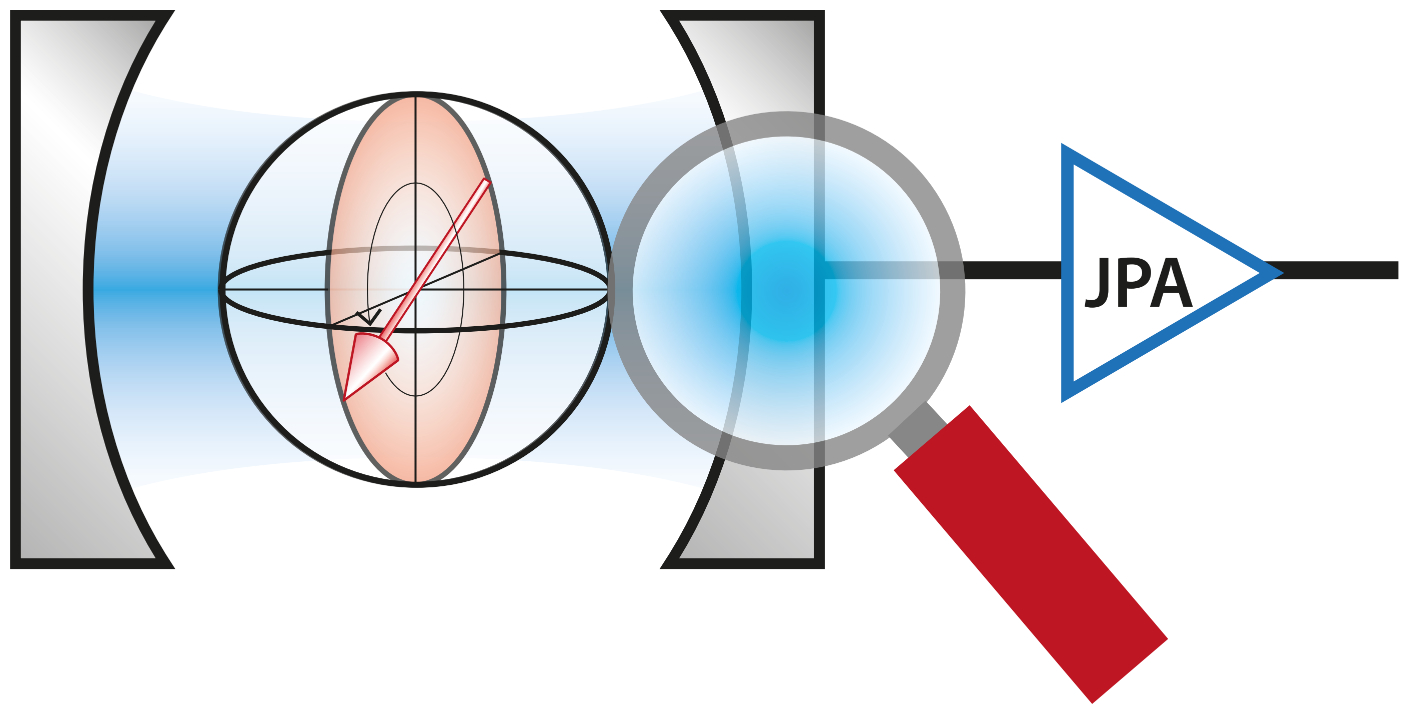


Dissipative evolution

$$\partial_t \rho = \left(i[\rho, H] + \sum_\sigma [\sigma^\dagger \rho \sigma - \frac{1}{2}(\sigma^\dagger \sigma \rho + \rho \sigma^\dagger \sigma)] \right) \mathbf{dt}$$

RNN

Parameter Estimation & Sensing



Dephasing rate

$\gamma = 0.82$ MHz

Rabi frequency

$\Omega_R = 1.10$ MHz

quantum efficiency

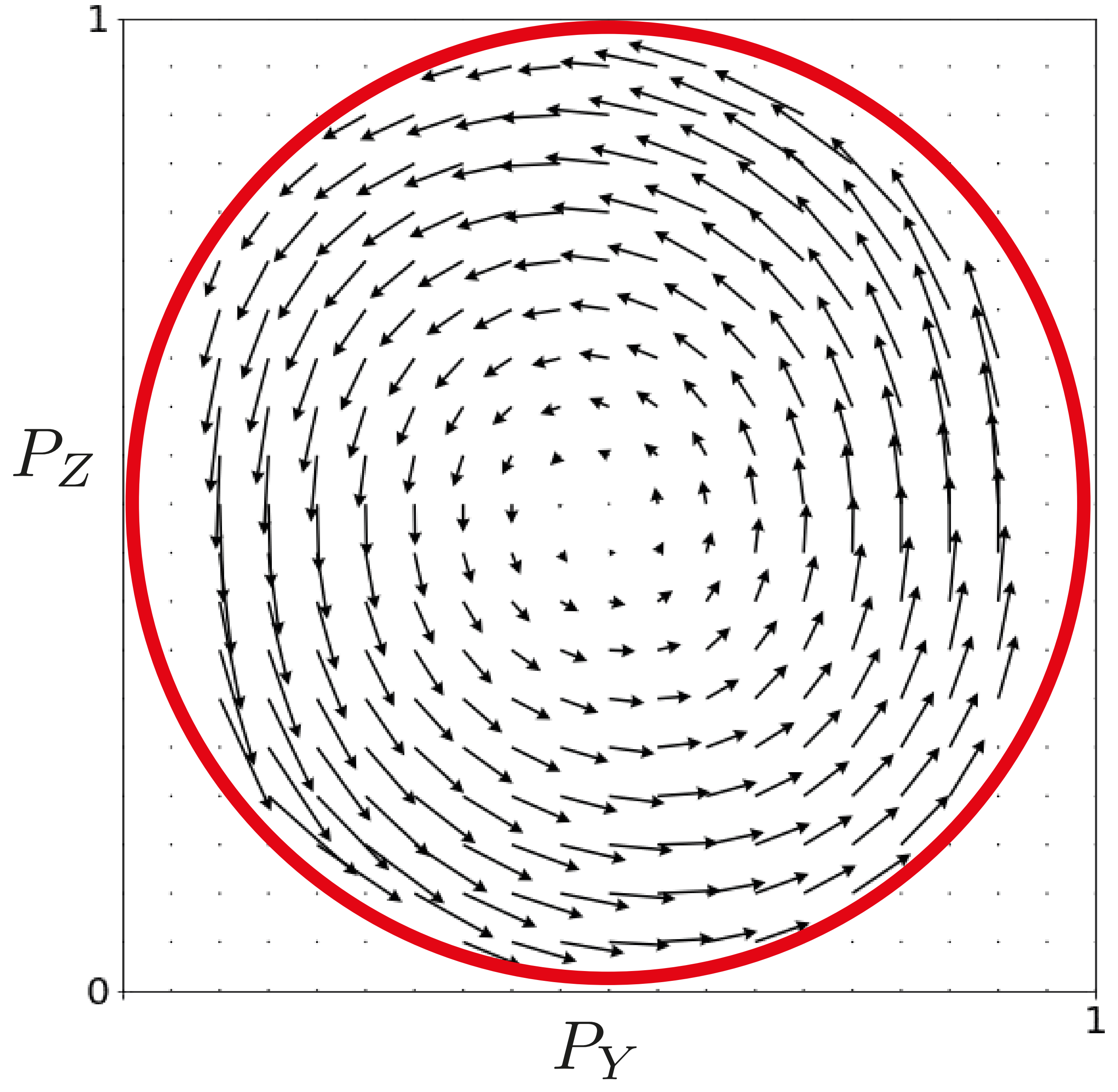
$\eta = 36\ \%$

$$\sigma = \sqrt{\gamma}\sigma_z$$

$$H = \Omega_R \frac{\sigma_x}{2}$$

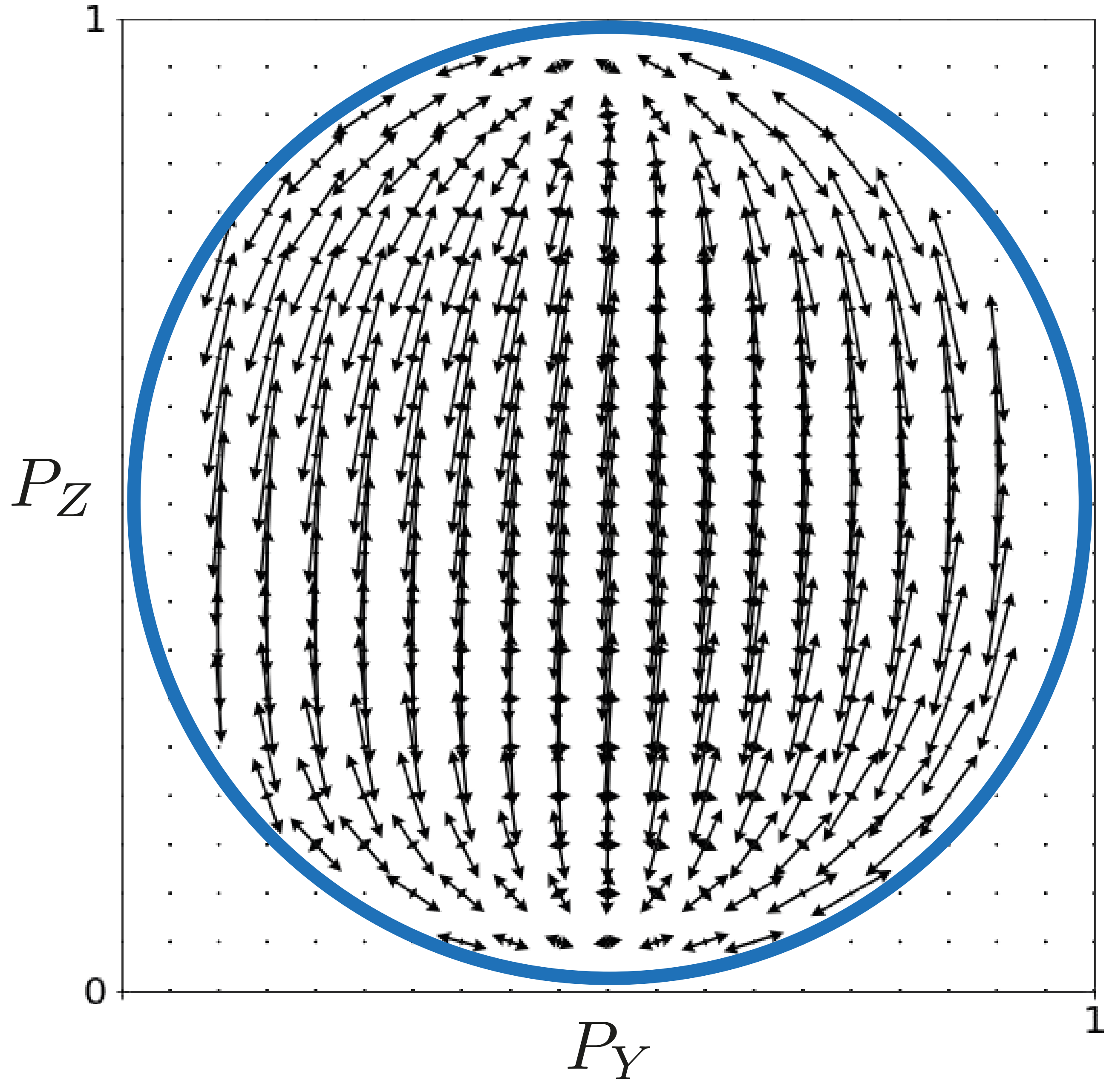
Drift map

$\langle \vec{P}_{t+dt} - \vec{P}_t \rangle$



Diffusion map

$\mathrm{covar}(\vec{P}_{t+dt} - \vec{P}_t)$



Dissipative evolution

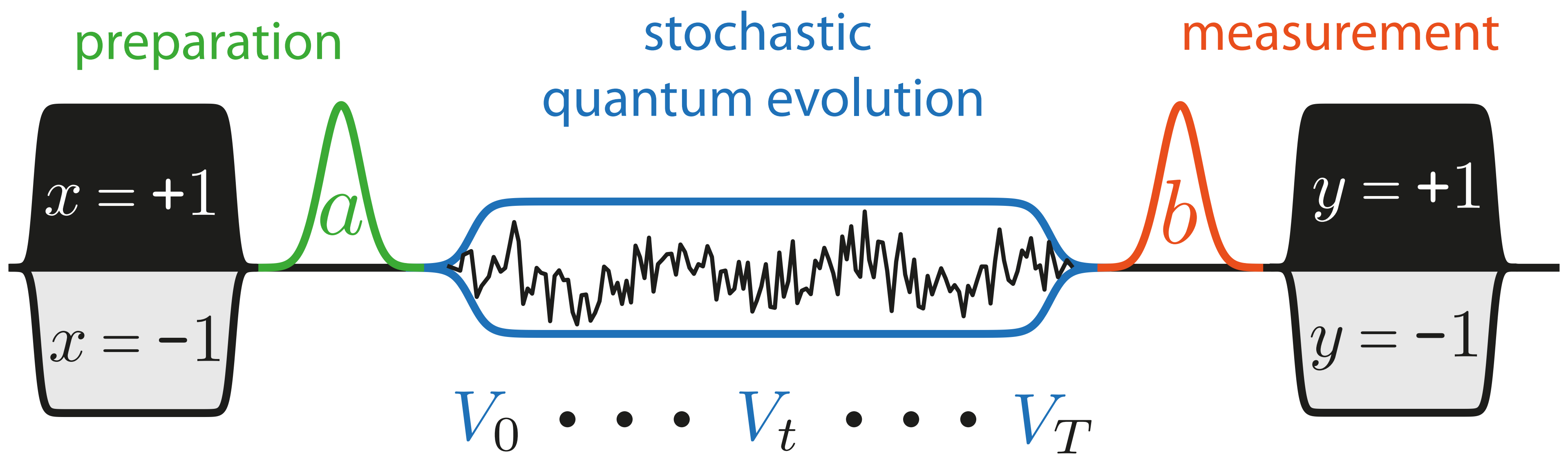
Measurement back-action

$$\partial_t \rho = \left(i[\rho, H] + \sum_\sigma [\sigma^\dagger \rho \sigma - \frac{1}{2}(\sigma^\dagger \sigma \rho + \rho \sigma^\dagger \sigma)]\right)\mathbf{dt} - \sqrt{\eta_\sigma}(\sigma\rho + \rho\sigma - 2\mathrm{Tr}(\sigma\rho)\rho)\mathbf{dW}$$

Prediction and retrodiction with bi-directional RNN

Prediction

$$P(y_t|x, a, b, V_0, \ldots, V_t)$$



Retrodiction

$$P(x|y_t, a, b, V_T, \ldots, V_t)$$

[The two-state vector formalism of quantum mechanics
Y Aharonov, L Vaidman (2002)]

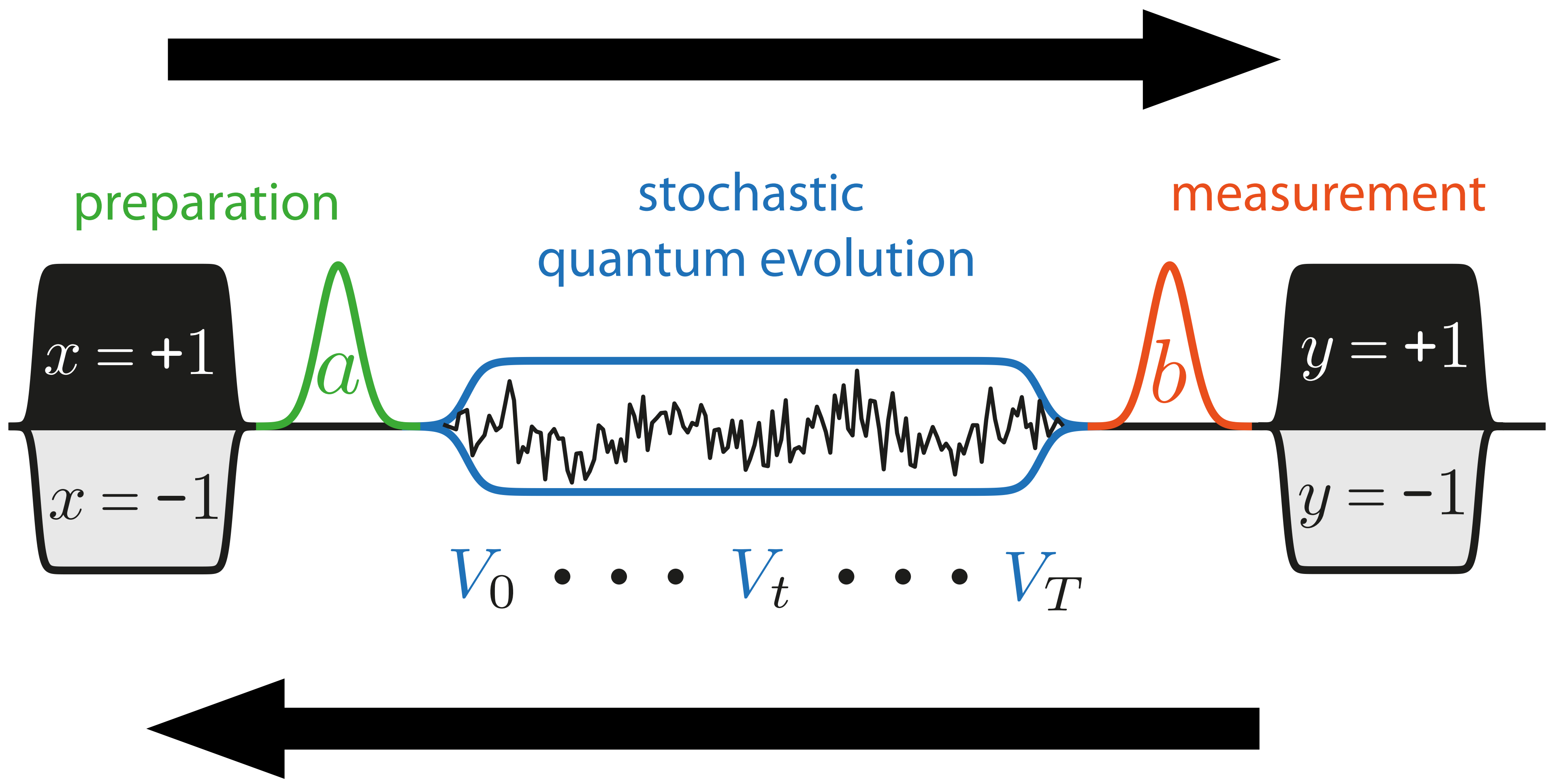
[Prediction and retrodiction for a continuously
monitored superconducting qubit
D. Tan, S. Weber, I. Siddiqi, K. Mølmer, K. W. Murch PRL (2015)]

bi-RNN

Prediction and retrodiction with bi-directional RNN

Prediction

$$P(y_t|x, a, b, V_0, \dots, V_t)$$



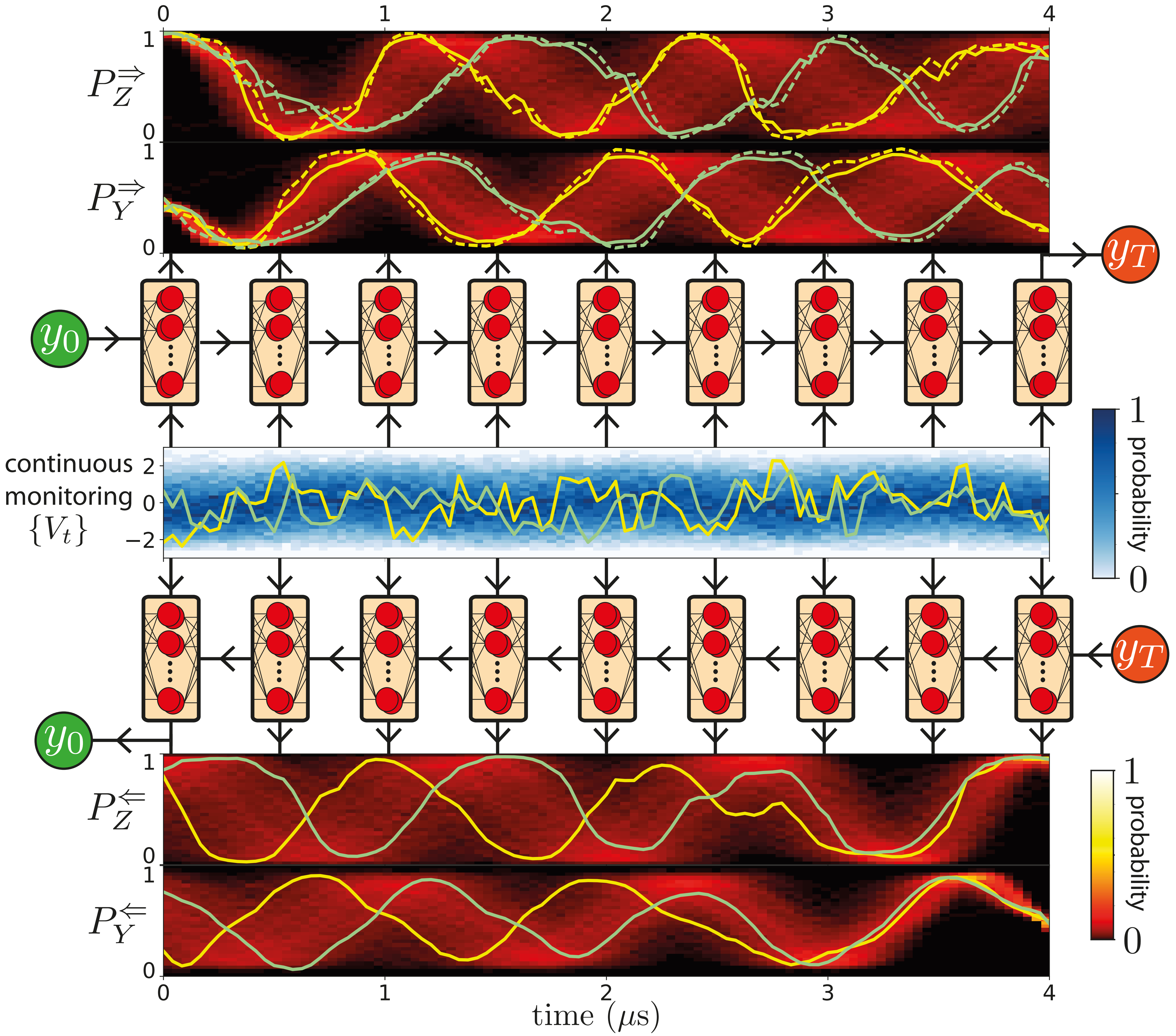
Retrodiction

$$P(x|y_t, a, b, V_T, \dots, V_t)$$

[The two-state vector formalism of quantum mechanics
Y Aharonov, L Vaidman (2002)]

[Prediction and retrodiction for a continuously
monitored superconducting qubit
D. Tan, S. Weber, I. Siddiqi, K. Mølmer, K. W. Murch PRL (2015)]

forward prediction



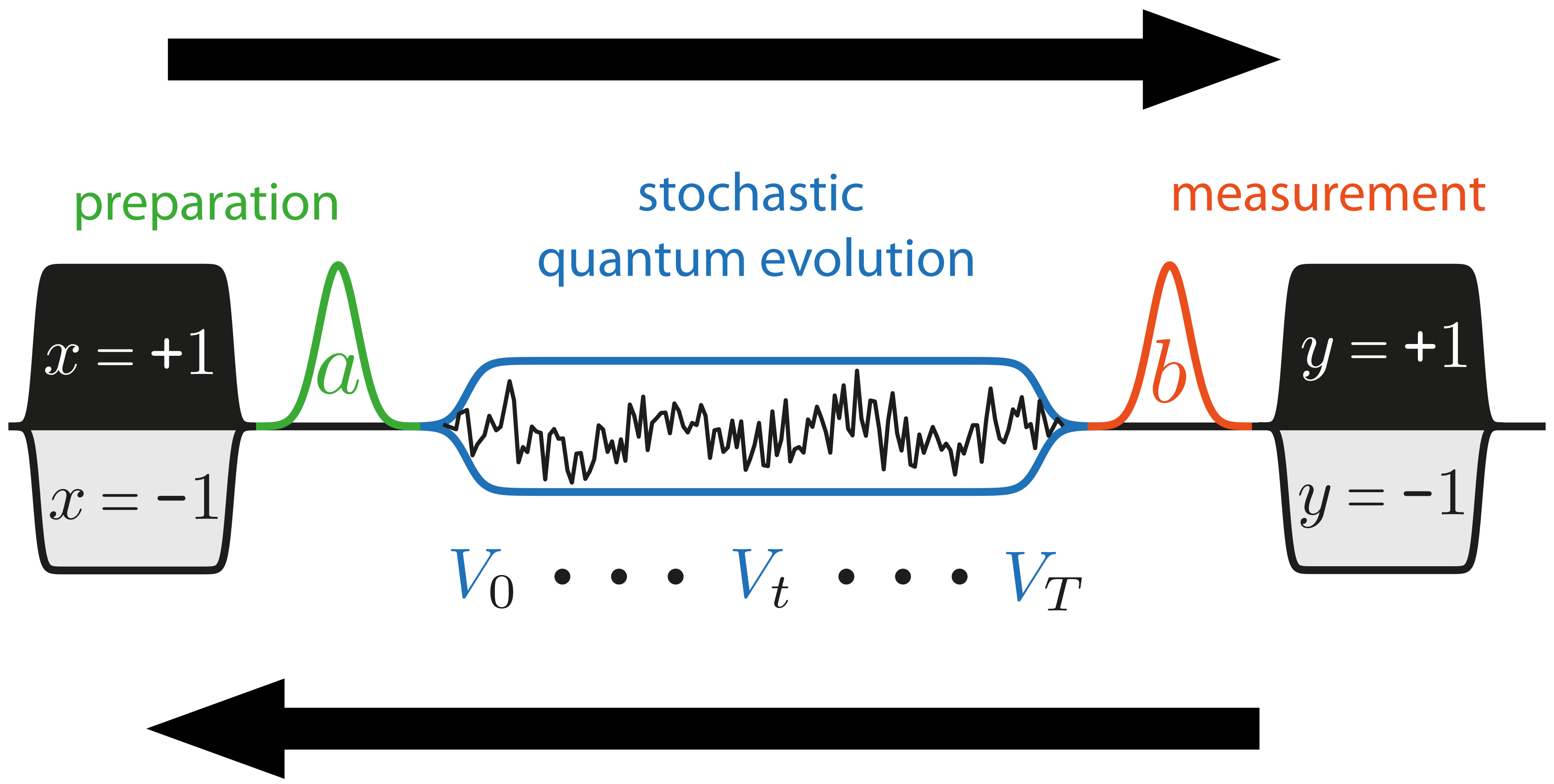
backward prediction

bi-RNN

Prediction and retrodiction with bi-directional RNN

Prediction

$$P(y_t | x, a, b, V_0, \ldots, V_t)$$

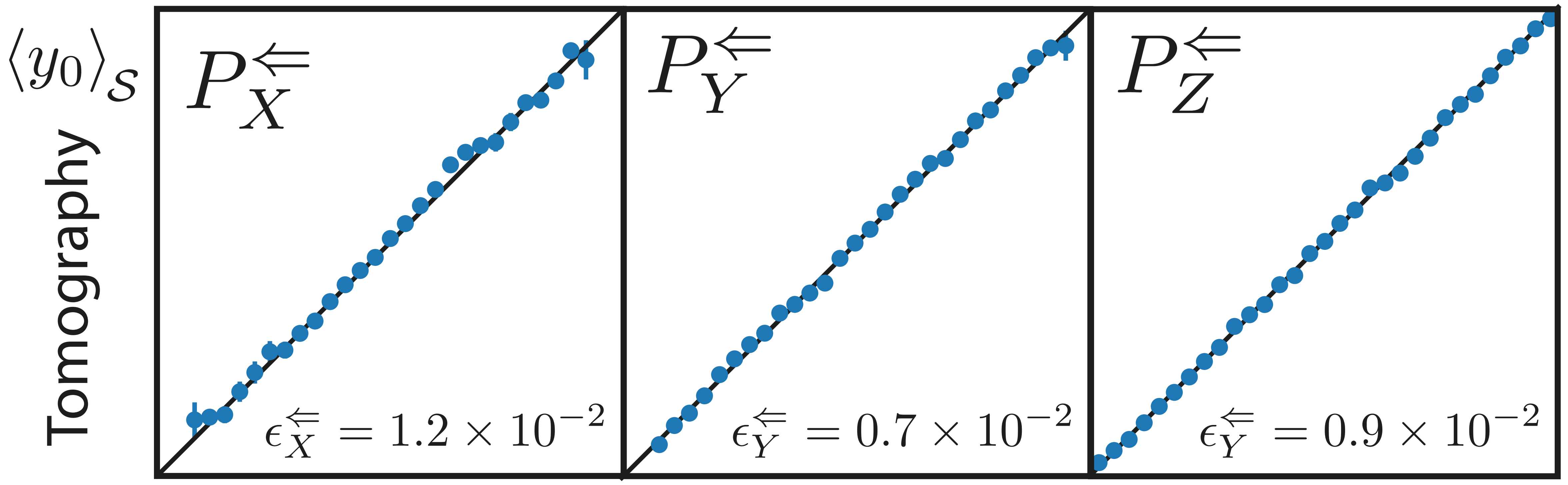


Retrodiction

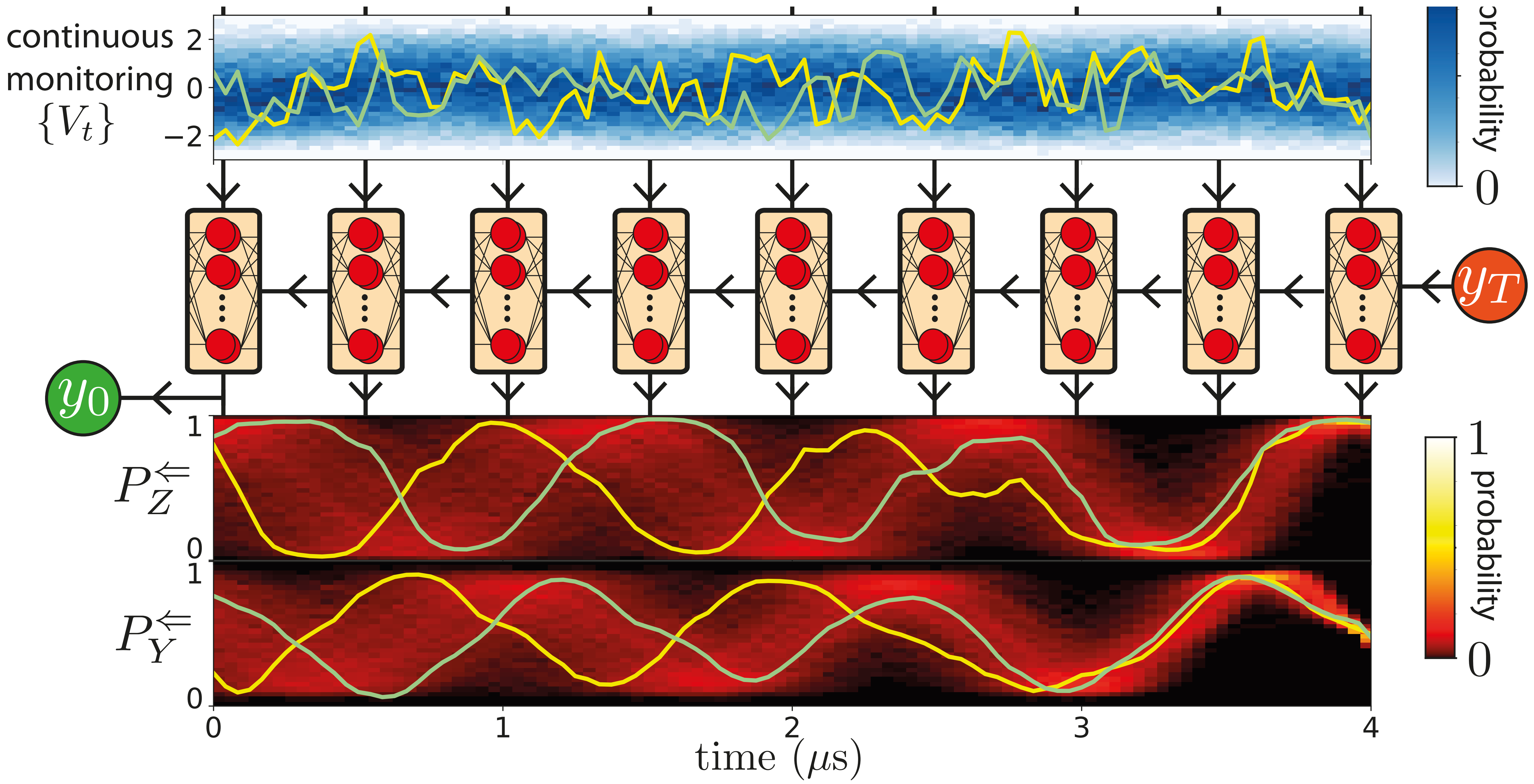
$$P(x | y_t, a, b, V_T, \ldots, V_t)$$

[The two-state vector formalism of quantum mechanics
Y Aharonov, L Vaidman (2002)]

[Prediction and retrodiction for a continuously
monitored superconducting qubit
D. Tan, S. Weber, I. Siddiqi, K. Mølmer, K. W. Murch PRL (2015)]



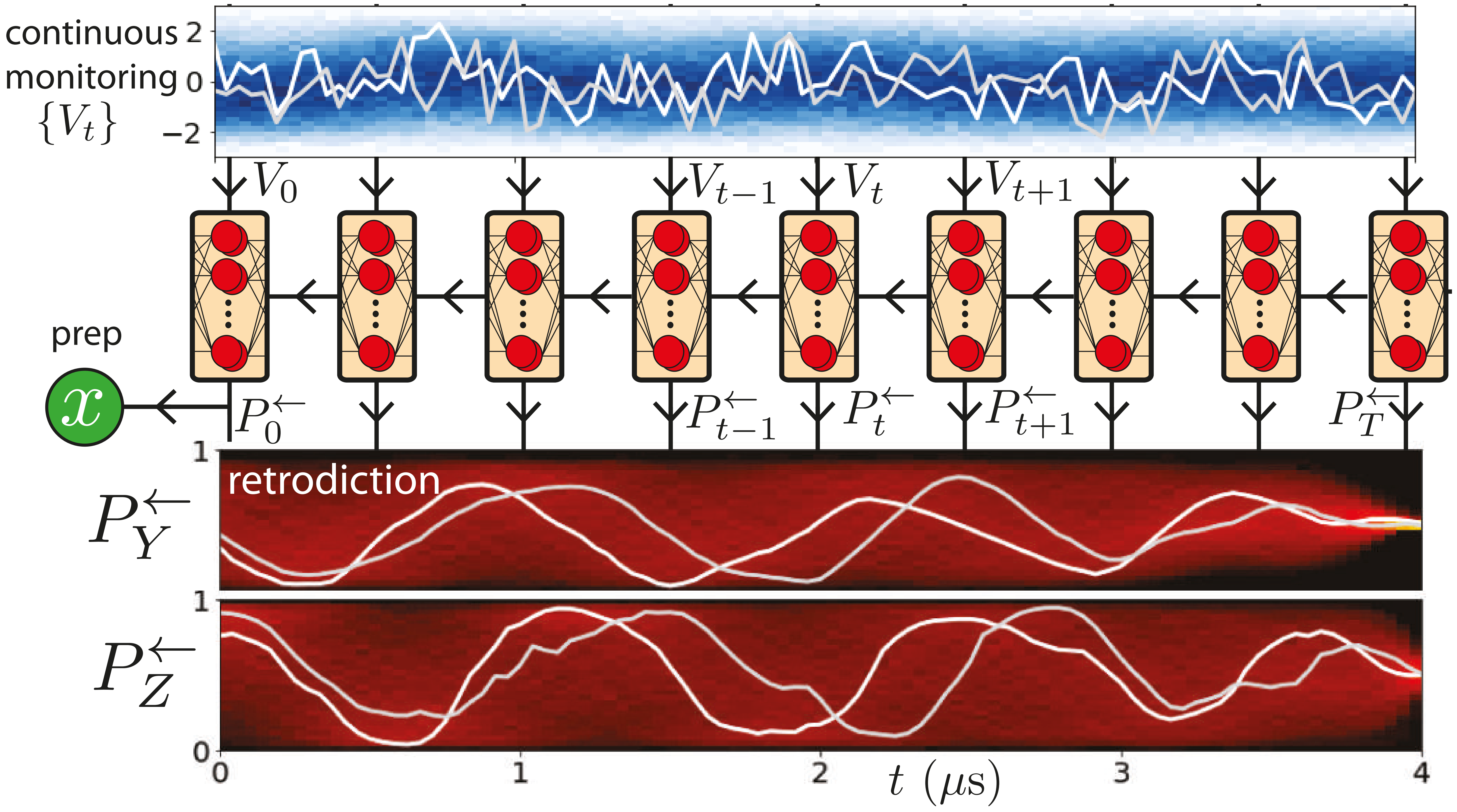
Backward Predictions



backward prediction

bi-RNN

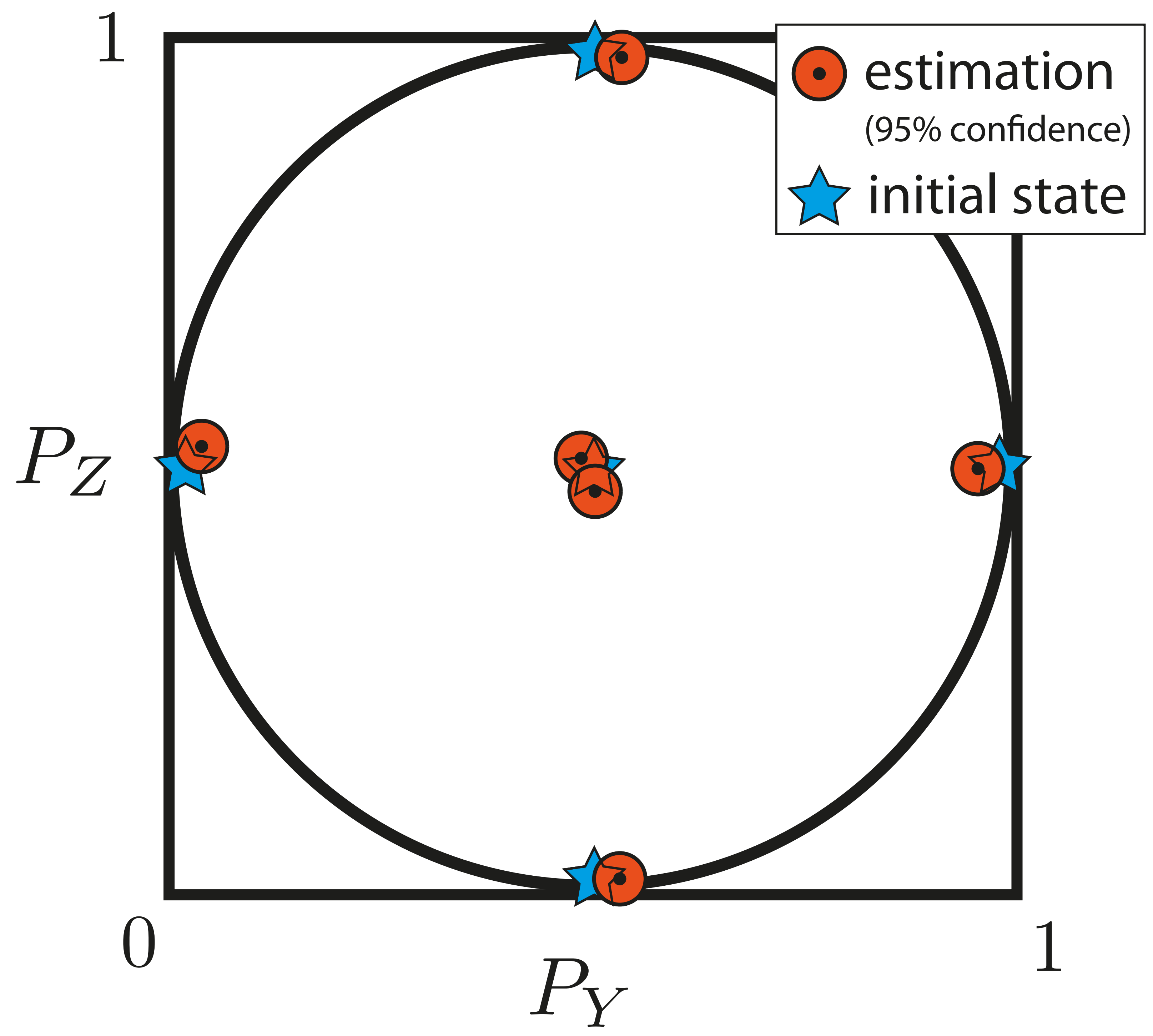
State Tomography from Retrodiction



Retrodiction

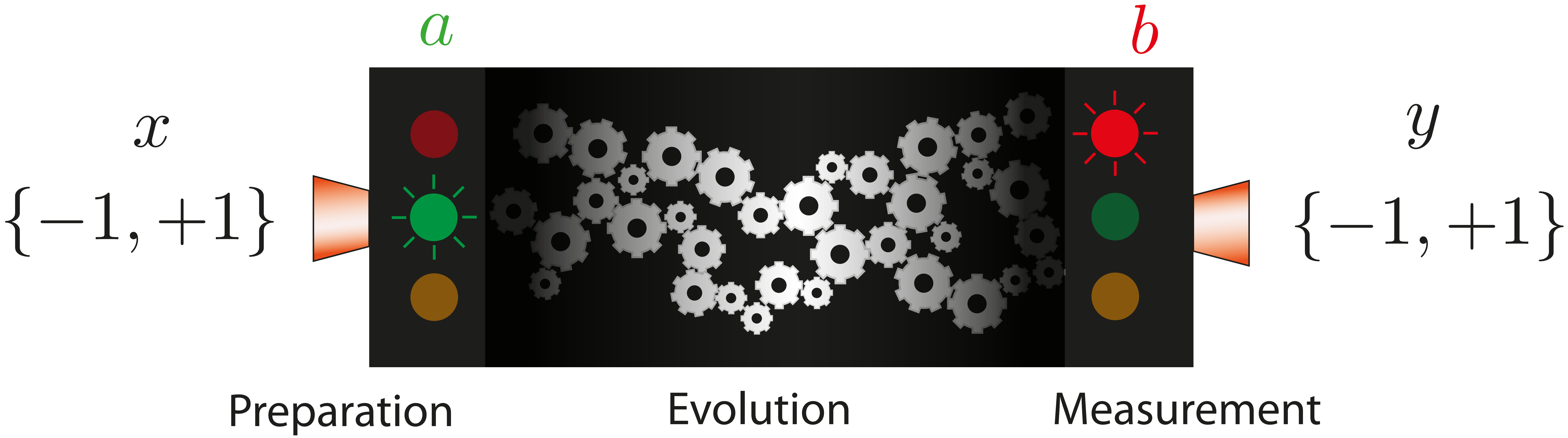
$$P(x|a, V_T, \ldots, V_t)$$

maximum likelihood reconstruction from 5000 trajectories



[Quantum state tomography with noninstantaneous measurements, imperfections, and decoherence
Six, Campagne-Ibarcq, Dotsenko, Sarlette, Huard, and Rouchon PRA (2016)]

Black Box Experiement

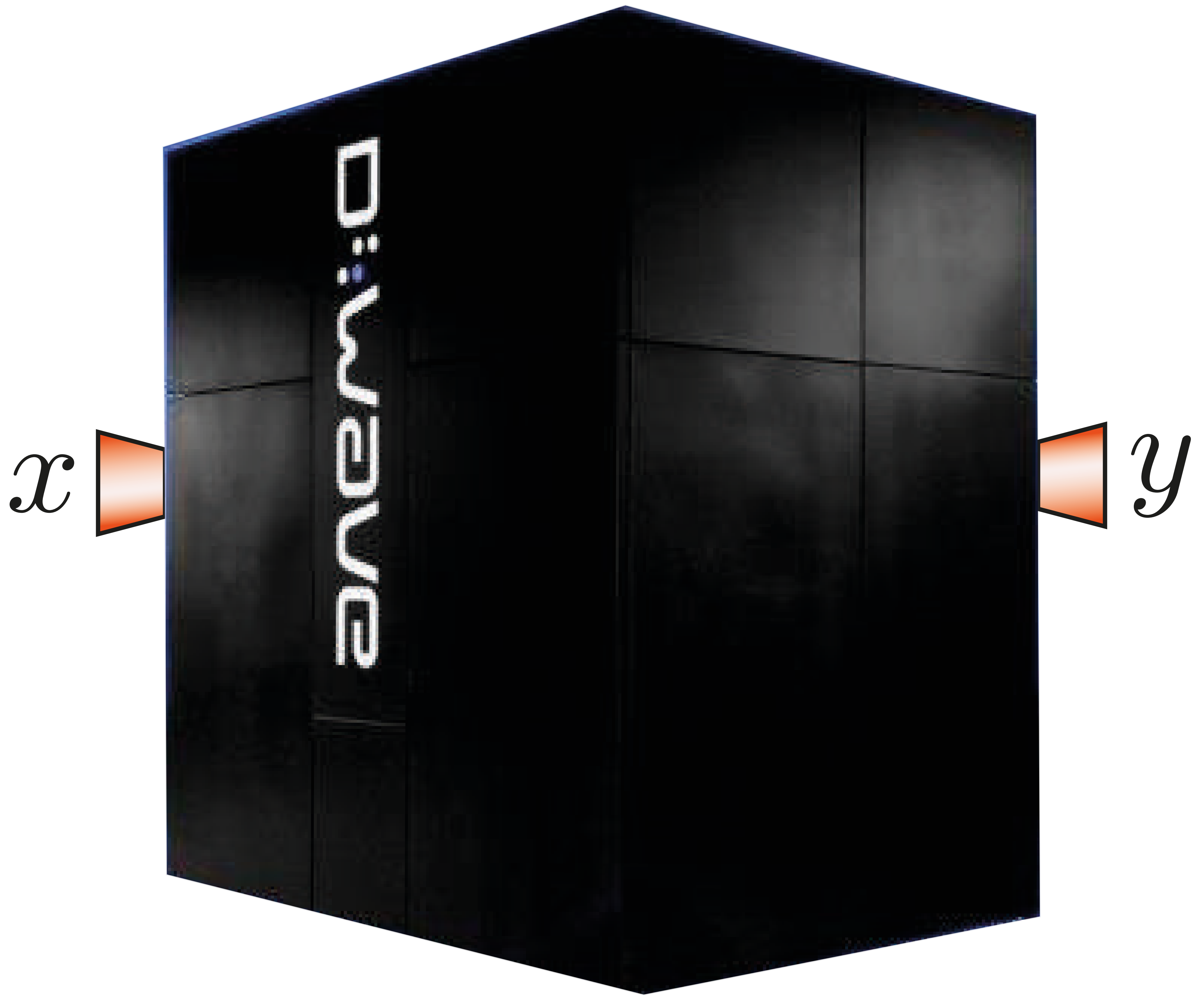


- **Model independent** validation of quantum trajectories beyond Markov approximation

- Efficient **extraction of physical parameters**
  - Quantum efficiency
  - Quantum state tomography

Big Quantum Machines

Black box quantum machines will require black box models.



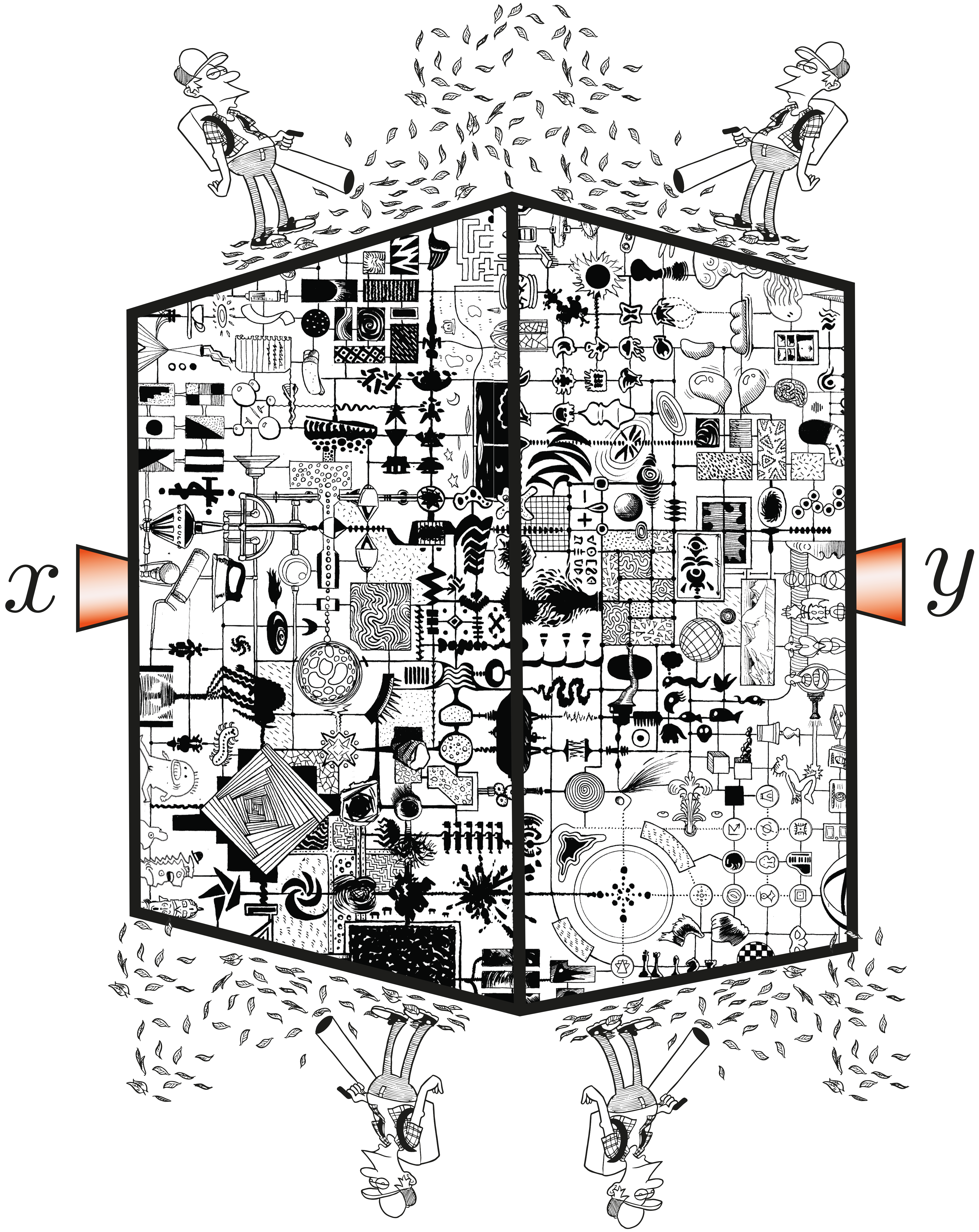
conclusion

Big Quantum Machines

Noise models are not precisely known a priori:

- non-markovian noise
- Correlated errors

Neural Networks can help to identify ***spatial and temporal*** hidden correlation



Efficient design and decoding of Quantum Error Correction

« Machine-learning-assisted correction of correlated qubit errors in a topological code »
Baireuther et al., arXiv

« Neural Decoder for Topological Codes »
Torlai et al., PRL 2018

« Reinforcement Learning with Neural Networks for Quantum Feedback »
Fösel et al., PRX (2018)

Quantum many-body problem

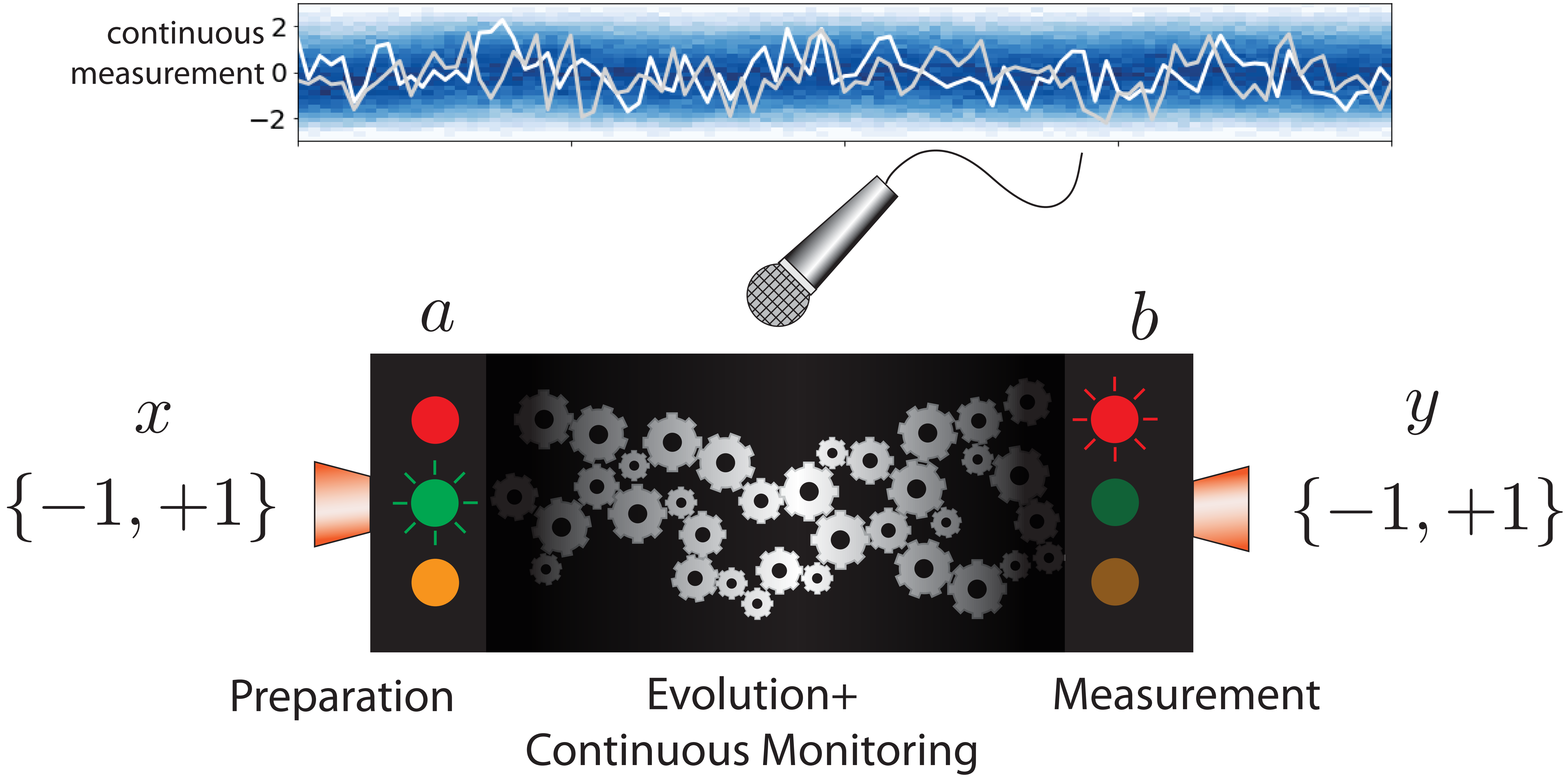
« Solving the quantum many-body problem with artificial neural networks »
Carlea, Troyer. Science (2017)

« Machine learning phases of matters »
Carrasquilla, Melko. Nature Physics (2017)

Thanks


S. Hacohen
-Gourgy
L.S. Martin
arXiv:1811.12420
continuous monitoring
$\{V_t\}$
V_0 V_{t-1} V_t V_{t+1} V_T
prep
P_0 P_{t-1} P_t P_{t+1} P_T
meas
P_Z
P_Y
P_X
CENTER for QUANTUM
COHERENT SCIENCE
Quantum
Nanoelectronics
Laboratory
THE UNIVERSITY OF CALIFORNIA BERKELEY
LET THERE BE LIGHT
1868

Quantum Master Equation



$$P(y|x, a, b, V_0, ..., V_t, ..., V_T) = \mathrm{Tr}(|y\rangle\langle y|\hat{B}\rho(T)\hat{B}^\dagger)$$

with stochastic master equation

$$\partial_t\rho = i[\rho, H]\mathrm{dt} + \sum_\sigma[\sigma^\dagger\rho\sigma - \frac{1}{2}(\sigma^\dagger\sigma\rho + \rho\sigma^\dagger\sigma)]\mathrm{dt} - \sqrt{\eta_\sigma}(\sigma\rho + \rho\sigma - 2\mathrm{Tr}(\sigma\rho)\rho)\mathrm{dW}$$

and $\rho(0) = \hat{A}\rho_x\hat{A}^\dagger$

Each of these terms has to be fine-tuned and separately calibrated