Approximating open quantum systems

Using Stinespring dilation

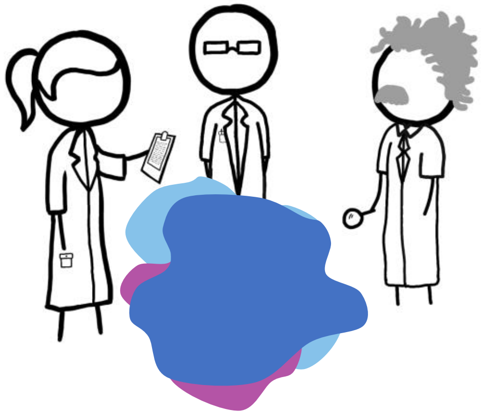
BFP Applied Physics, Applied Mathematics
David Chen, 1742477

My project in one minute

Create a model

(Most scientists)

Compare to experiment

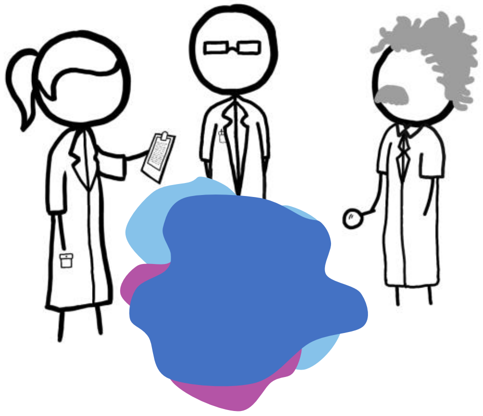


My project in one minute

Create a model

Compare to experiment

(Most scientists)



Infer model

Make measurements

(Me)

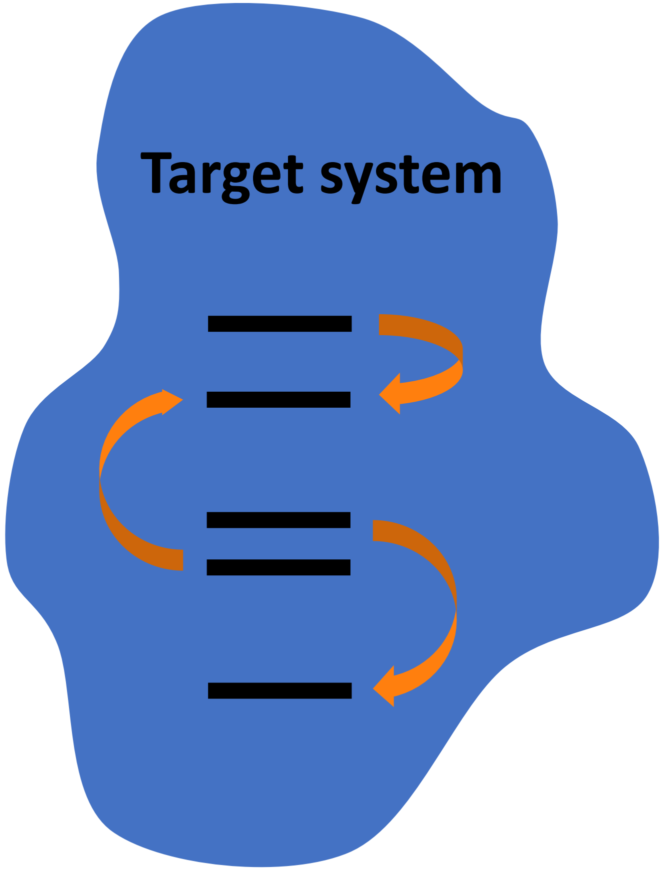
Menu

Background

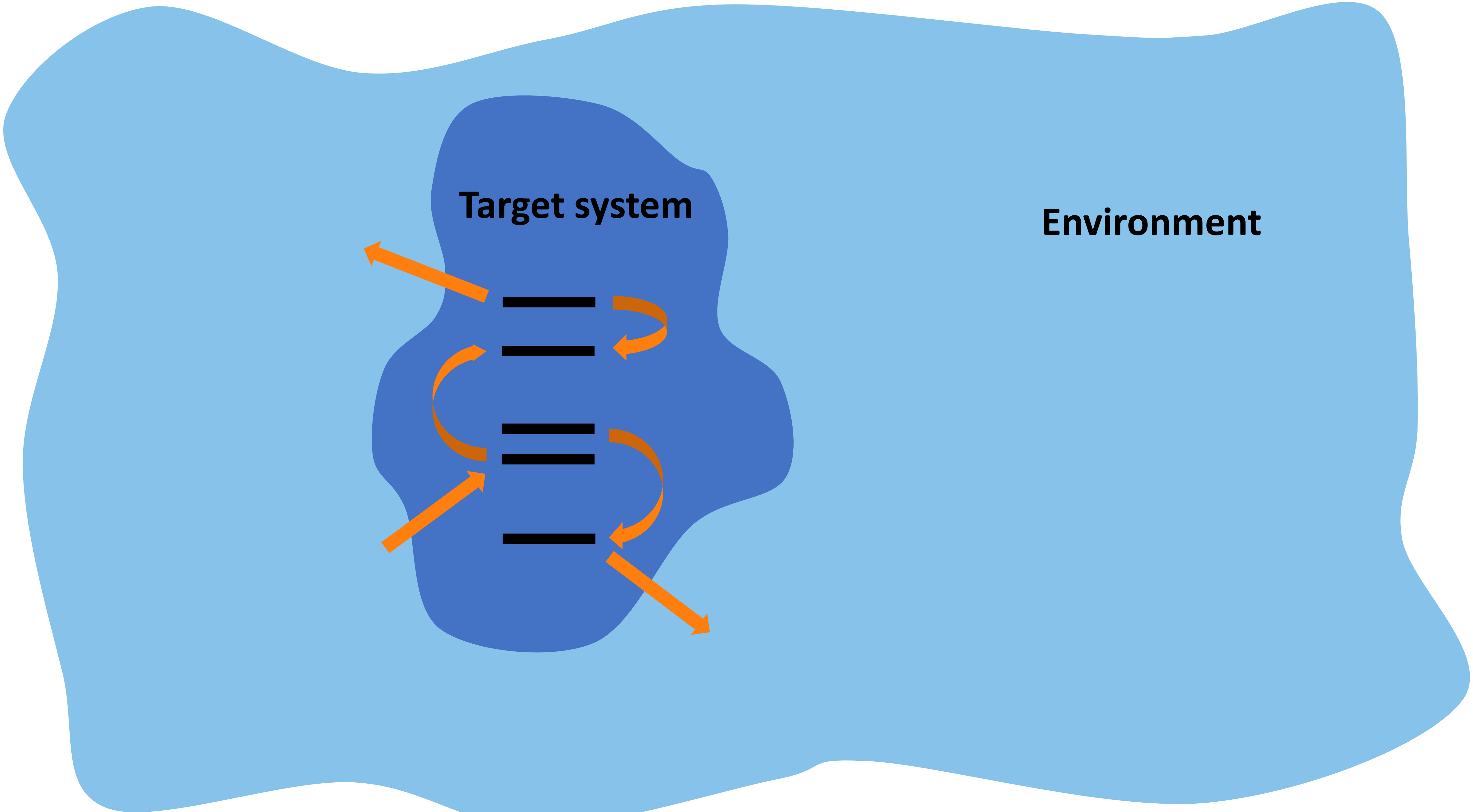
- Quantum systems
- Density matrix
- Evolution equations
- Quantum channels
- Combining qubits
- Stinespring's dilation theorem

My project

What is a quantum system?



What is an open quantum system?



Density matrix

$$|\psi\rangle = \alpha_n \left|e^n\right\rangle \xrightarrow{\text{Density matrix formalism}} \varrho = |\psi\rangle\langle\psi|$$

Pure state

$$= \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{bmatrix}$$

$$= \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{bmatrix} \begin{bmatrix} \overline{\alpha}_1 & \dots & \overline{\alpha}_N \end{bmatrix} = \begin{bmatrix} \alpha_1\overline{\alpha}_1 & \dots & \alpha_1\overline{\alpha}_N \\ \vdots & \ddots & \vdots \\ \alpha_N\overline{\alpha}_1 & \dots & \alpha_N\overline{\alpha}_N \end{bmatrix}$$

$$\rho = \sum_k p_k |\psi^k\rangle\langle\psi^k|, \quad \sum_k p_k = 1$$

Mixed state

Density matrix

$$|\psi\rangle = \alpha_n \, |e^n\rangle$$

$$= \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{bmatrix}$$

Density matrix formalism

N

$$\varrho = |\psi\rangle\langle\psi|$$

Pure state

$$= \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{bmatrix} \begin{bmatrix} \overline{\alpha}_1 & \dots & \overline{\alpha}_N \end{bmatrix} = \begin{bmatrix} \alpha_1\overline{\alpha}_1 & \dots & \alpha_1\overline{\alpha}_N \\ \vdots & \ddots & \vdots \\ \alpha_N\overline{\alpha}_1 & \dots & \alpha_N\overline{\alpha}_N \end{bmatrix}$$

$$\rho = \sum_k p_k |\psi^k\rangle\langle\psi^k|, \quad \sum_k p_k = 1$$

Mixed state

Evolution equations

Closed system (Schrödinger equation)

$$\frac{d}{dt}|\psi\rangle = -\frac{i}{\hbar}H|\psi\rangle$$

Density matrix formalism

N

(Von Neumann equation)

$$\frac{d}{dt}\rho = -\frac{i}{\hbar}[H,\rho]$$

Open system (Lindblad equation)

$$\frac{d}{dt}\rho = -\frac{i}{\hbar}[H,\rho] + \sum_k \gamma_k \left(A_k \rho A_k^\dagger - \frac{1}{2}\{A_k^\dagger A_k, \rho\} \right)$$

Evolution equations

Closed system (Schrödinger equation)

(Von Neumann equation)

$$\frac{d}{dt}|\psi\rangle = -\frac{i}{\hbar}H|\psi\rangle$$

Density matrix formalism $\longrightarrow$

$$\frac{d}{dt}\rho = -\frac{i}{\hbar}[H,\rho]$$

"Hamiltonian"

Open system (Lindblad equation)

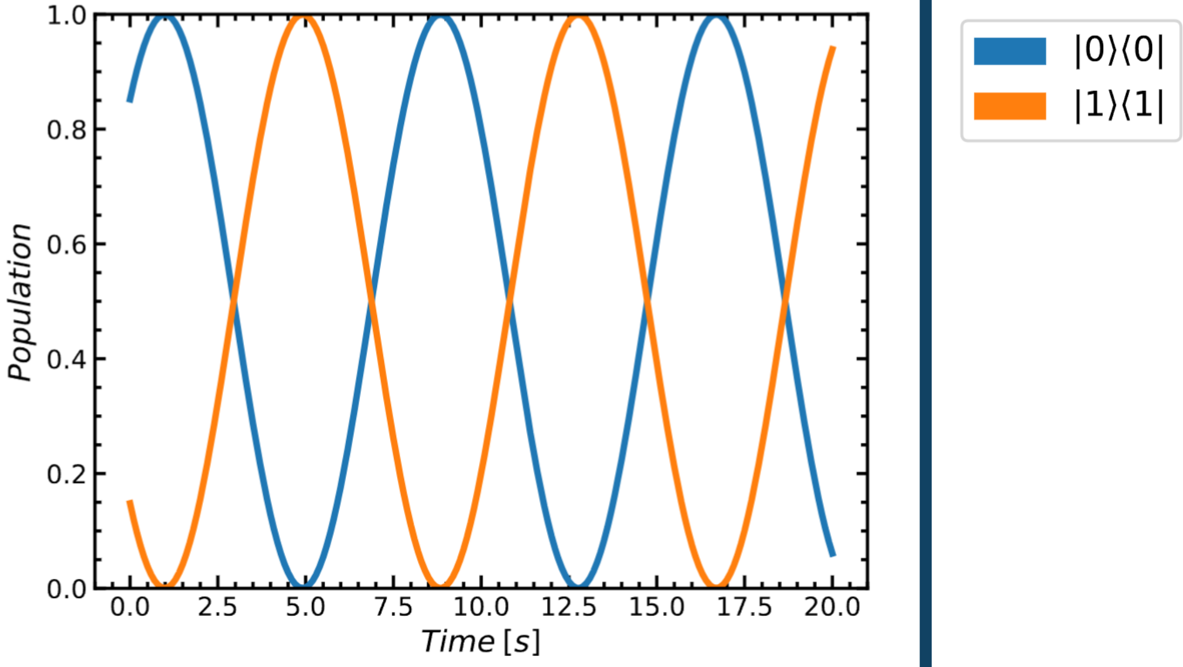
$$\frac{d}{dt}\rho = -\frac{i}{\hbar}[H,\rho] + \sum_k \gamma_k \left(A_k \rho A_k^\dagger - \frac{1}{2}\{A_k^\dagger A_k, \rho\} \right)$$

"Hamiltonian" "Interactions"

Evolution equations

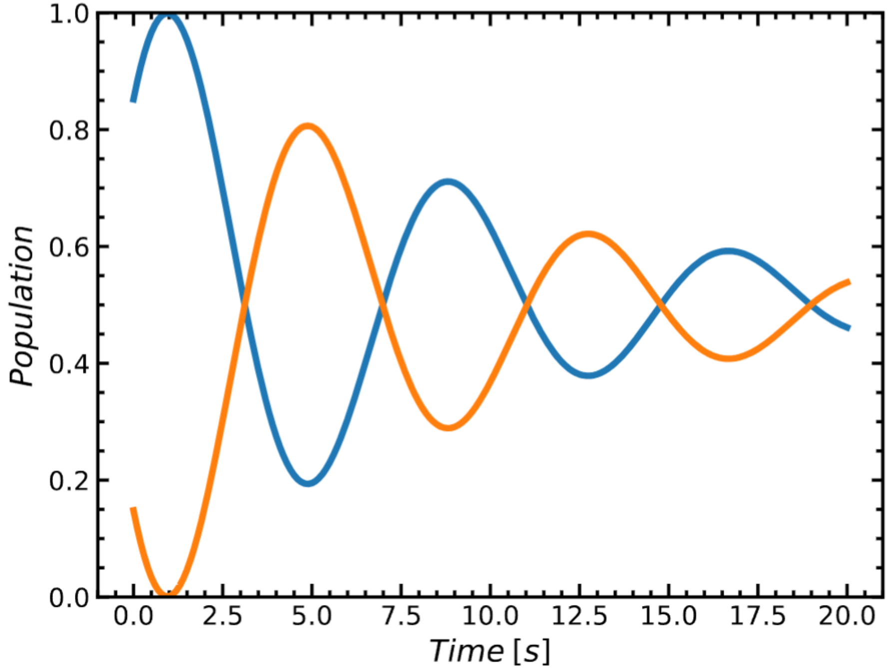
Closed system (Von Neumann)

$$\frac{d}{dt}\rho = -\frac{i}{\hbar}[H,\rho]$$



Open system (Lindblad)

$$\frac{d}{dt}\rho = -\frac{i}{\hbar}[H,\rho] + \sum_k \gamma_k \left(A_k \rho A_k^\dagger - \frac{1}{2}\{A_k^\dagger A_k, \rho\} \right)$$



Quantum channels

Closed system (Von Neumann)

$$\frac{d}{dt}\rho = -\frac{i}{\hbar}[H,\rho]$$

$$U_t \rho_0 U_t^\dagger = \rho_t,$$

$$U_t = \exp\left(-\frac{i}{\hbar}Ht\right)$$

Open system (Lindblad)

$$\frac{d}{dt}\rho = -\frac{i}{\hbar}[H,\rho] + \sum_k \gamma_k \left(A_k \rho A_k^\dagger - \frac{1}{2}\{A_k^\dagger A_k, \rho\}\right)$$

Goal: approximate $\Phi_t(\rho_0) = \rho_t$

Quantum channels

Closed system (Von Neumann)

$$\frac{d}{dt}\rho = -\frac{i}{\hbar}[H,\rho]$$

$$U_t \rho_0 U_t^\dagger = \rho_t,$$

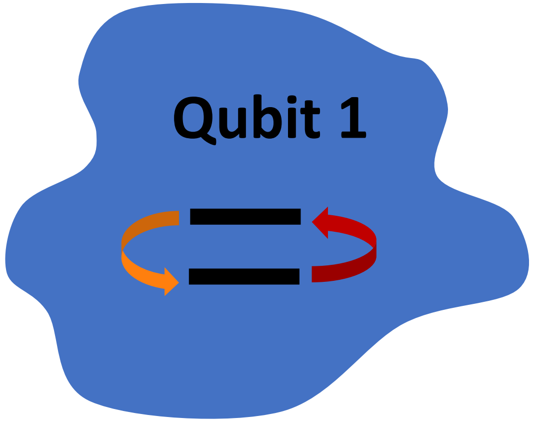
$$U_t = \exp\left(-\frac{i}{\hbar}Ht\right)$$

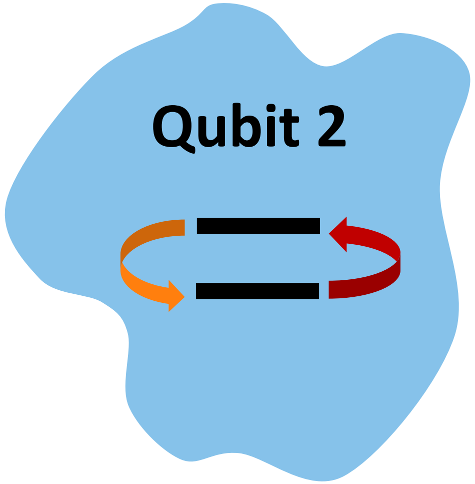
Open system (Lindblad)

$$\frac{d}{dt}\rho = -\frac{i}{\hbar}[H,\rho] + \sum_k \gamma_k \left(A_k \rho A_k^\dagger - \frac{1}{2}\{A_k^\dagger A_k, \rho\}\right)$$

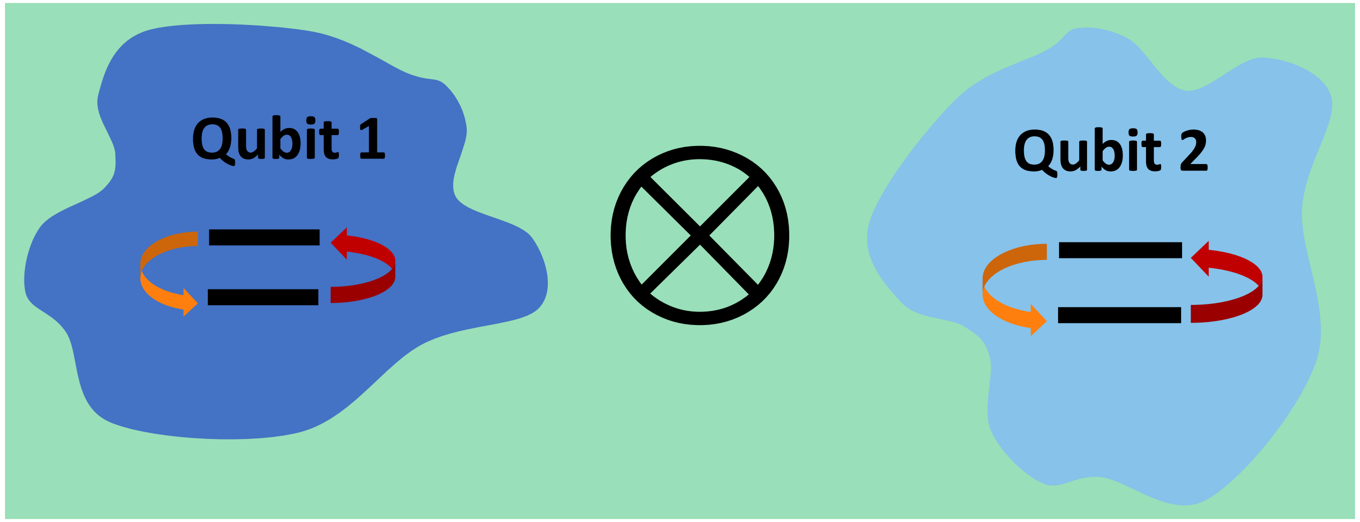
Goal: approximate $\Phi_t(\rho_0) = \rho_t$

Combining qubits



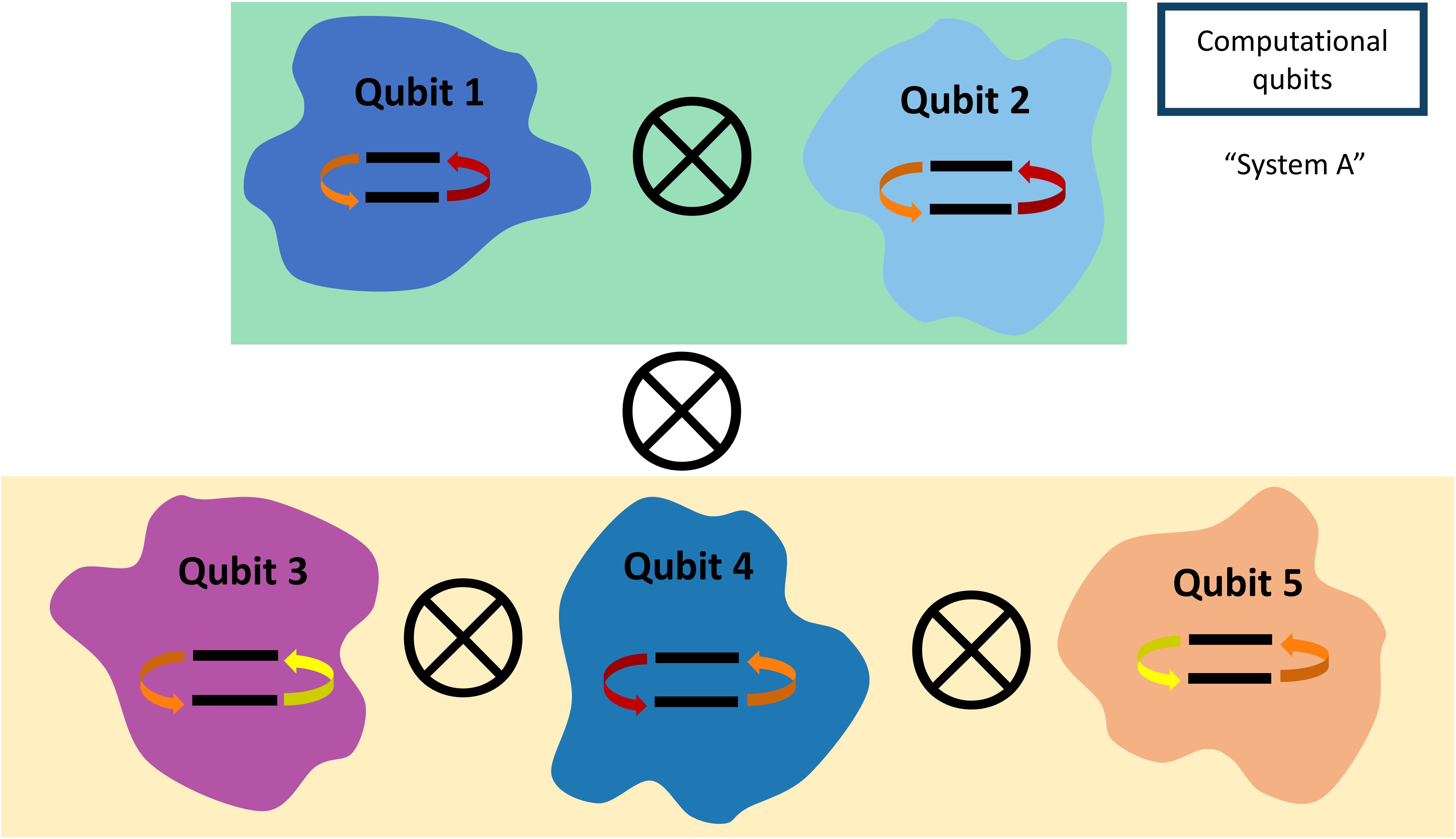


Combining qubits



Computational qubits

Combining qubits



Computational qubits

"System A"

Ancilla qubits

"System B"

Stinespring's dilation theorem

There exists U_t on dilated space (computational qubits) $\otimes$ (ancilla qubits) such that:

$$\Phi_t(\rho_0) = \mathrm{Tr}_B[U_t\,(\rho_0 \otimes |0\rangle_B\langle 0|_B)\,U_t^\dagger]$$

Stinespring's dilation theorem

There exists U_t on dilated space (computational qubits) $\otimes$ (ancilla qubits) such that:

$$\Phi_t(\rho_0) = \mathrm{Tr}_B[U_t\,(\rho_0 \otimes |0\rangle_B\langle 0|_B)\,U_t^\dagger]$$

"Stinespring unitary"

Menu

My project

- Approximation method
- Parametrization of Stinespring unitary
- Results

Approximation method

Gather training data

(Initial state) ρ_0 → Evolve for time t → ρ_t → Measure expectations of $O_1, O_2, \ldots$ → (Expectation values) $E_1, E_2, \ldots$

Repeat for different initial states

Approximate quantum channel

$$\Phi_t(\rho_0) = \mathrm{Tr}_B[U_t\,(\rho_0 \otimes |0\rangle_B\langle 0|_B)\,U_t^\dagger]$$

$\Phi_t(\rho_0) \approx \rho_t$

Optimize parameters

$$J(\theta) = \sum_\ell (E_\ell - \hat{E}_\ell)^2$$

Approximation method

Gather training data

(Initial state)

ρ_0

Evolve for time t

ρ_t

Measure expectations of $O_1, O_2, \ldots$

(Expectation values)

$E_1, E_2, \ldots$

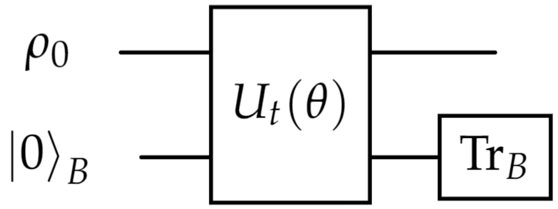
Repeat for different initial states

Approximate quantum channel

$$\Phi_t(\rho_0) = \mathrm{Tr}_B[U_t\,(\rho_0 \otimes |0\rangle_B\langle 0|_B)\,U_t^\dagger]$$

Parametrize Stinespring unitary

“Quantum circuit”



$\Phi_t(\rho_0) \approx \rho_t$

Optimize parameters

$$J(\theta) = \sum_\ell (E_\ell - \hat{E}_\ell)^2$$

Approximation method

Gather training data

(Initial state) ρ_0 $\xrightarrow{\text{Evolve for time } t}$ ρ_t $\xrightarrow{\text{Measure expectations of } O_1, O_2, \ldots}$ (Expectation values) $E_1, E_2, \ldots$

Evolve for time t

Measure expectations of $O_1, O_2, \ldots$

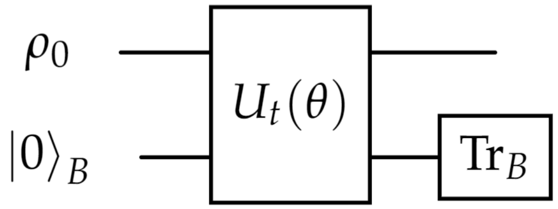
Repeat for different initial states

Approximate quantum channel

$$\Phi_t(\rho_0) = \mathrm{Tr}_B[U_t\,(\rho_0 \otimes |0\rangle_B \langle 0|_B)\,U_t^\dagger]$$

Parametrize Stinespring unitary

“Quantum circuit”



$\Phi_t(\rho_0) \approx \rho_t$

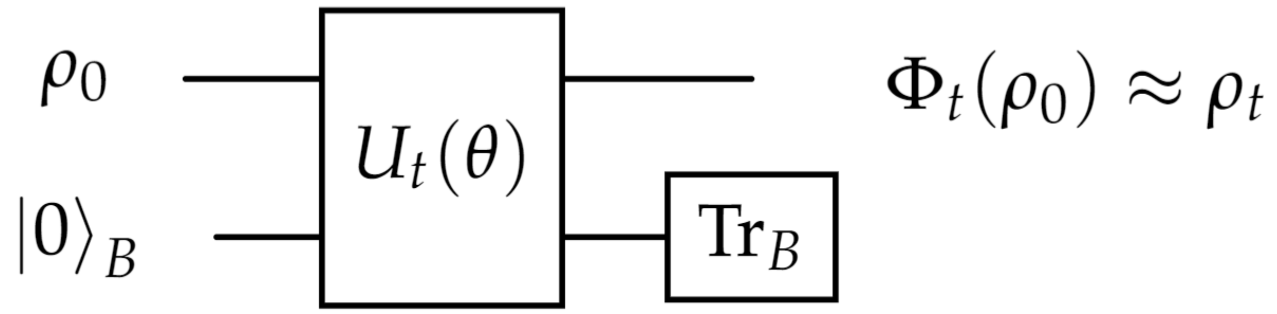
Optimize parameters

$$J(\theta) = \sum_\ell (E_\ell - \hat{E}_\ell)^2$$

Measured

Approximated

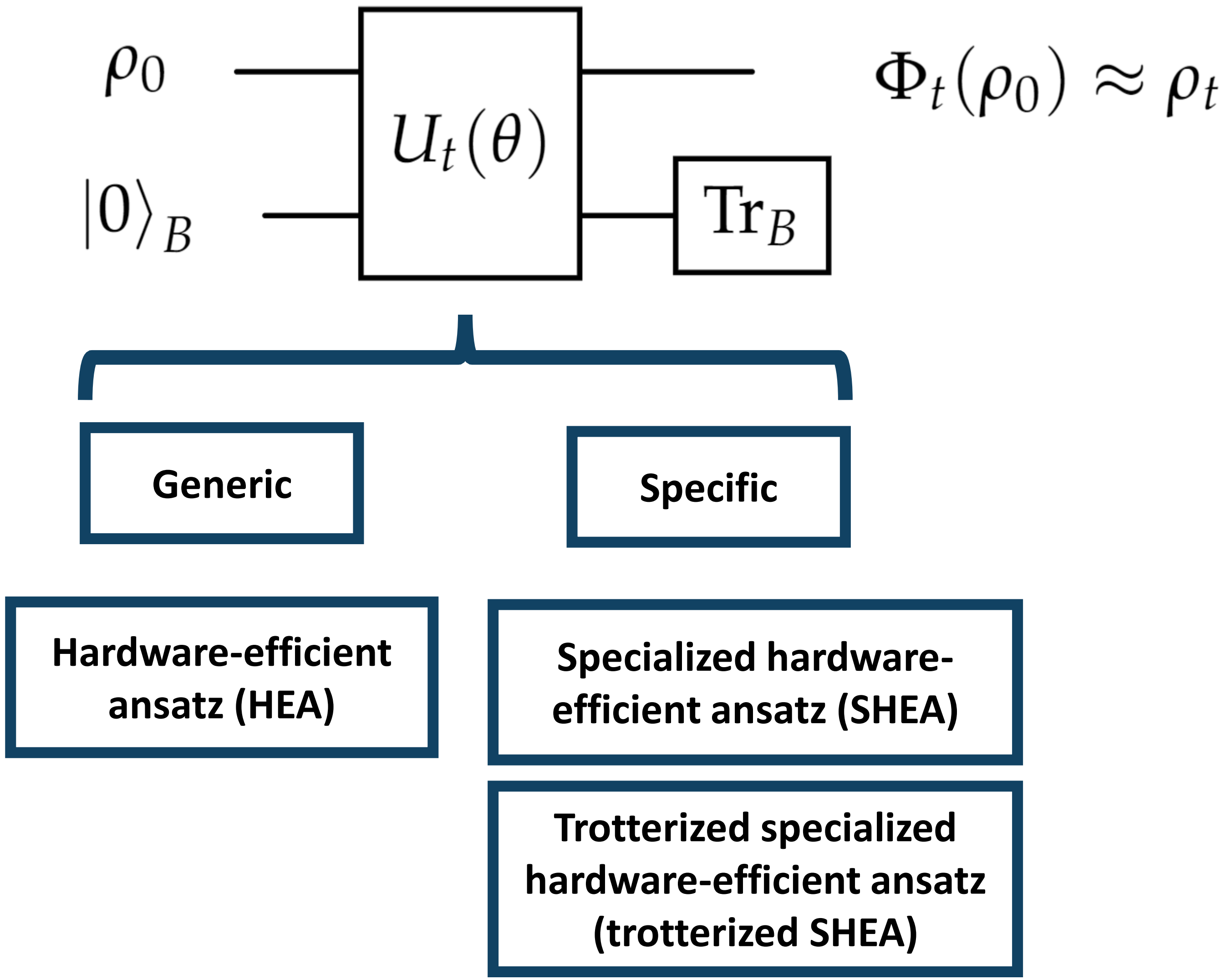
Parametrization of Stinespring unitary



Generic

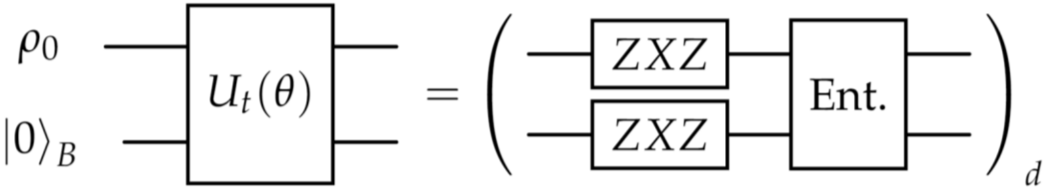
Hardware-efficient ansatz (HEA)

Parametrization of Stinespring unitary



Parametrization of Stinespring unitary

Hardware-efficient ansatz (HEA)



Specialized hardware-efficient ansatz (SHEA)

Interlude: trotterization

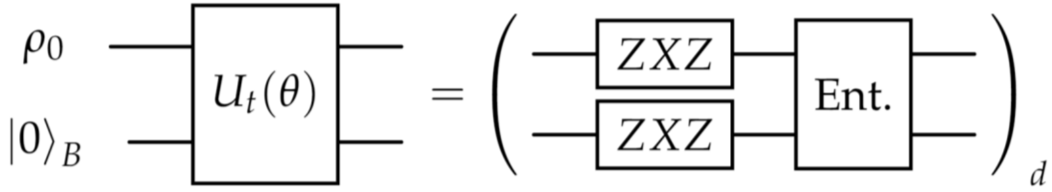
For Hermitian operators A and B we have:

$$\lim_{n\to\infty}\left(e^{-iA\frac{t}{n}}e^{-iB\frac{t}{n}}\right)^n = e^{-i(A+B)t}$$

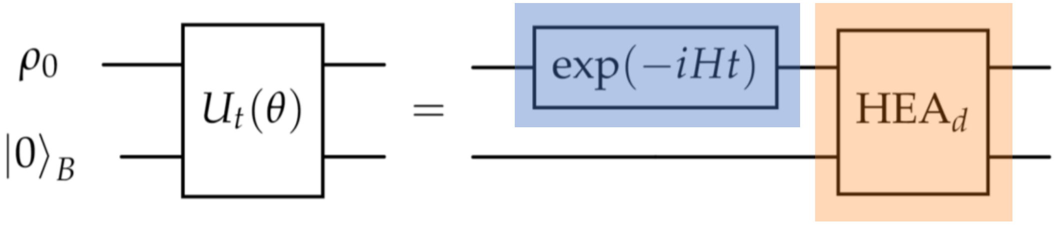
Trotterized specialized hardware-efficient ansatz (trotterized SHEA)

Parametrization of Stinespring unitary

Hardware-efficient ansatz (HEA)



Specialized hardware-efficient ansatz (SHEA)



$$\frac{d}{dt}\rho = -\frac{i}{\hbar}[H,\rho] + \sum_k \gamma_k \left(A_k \rho A_k^\dagger - \frac{1}{2}\{A_k^\dagger A_k, \rho\}\right)$$

Interlude: trotterization

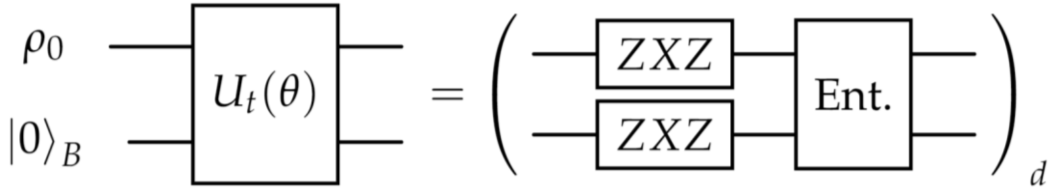
For Hermitian operators A and B we have:

$$\lim_{n\to\infty} \left(e^{-iA\frac{t}{n}} e^{-iB\frac{t}{n}}\right)^n = e^{-i(A+B)t}$$

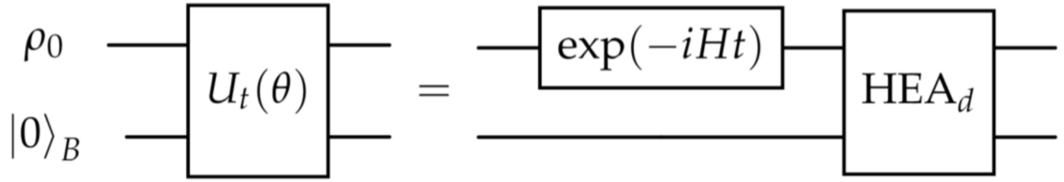
Trotterized specialized hardware-efficient ansatz (trotterized SHEA)

Parametrization of Stinespring unitary

Hardware-efficient ansatz (HEA)



Specialized hardware-efficient ansatz (SHEA)



$$\frac{d}{dt}\rho = -\frac{i}{\hbar}[H,\rho] + \sum_k \gamma_k \left(A_k \rho A_k^\dagger - \frac{1}{2}\{A_k^\dagger A_k, \rho\} \right)$$

Interlude: trotterization

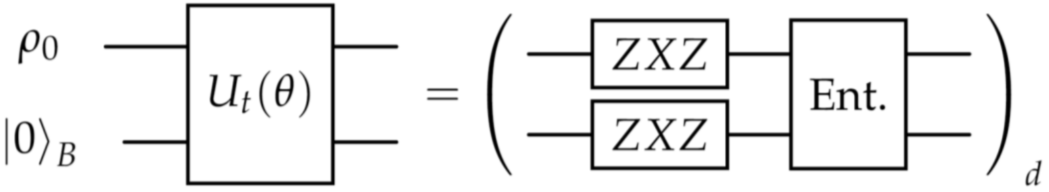
For Hermitian operators A and B we have:

$$\lim_{n\to\infty} \left(e^{-iA\frac{t}{n}} e^{-iB\frac{t}{n}}\right)^n = e^{-i(A+B)t}$$

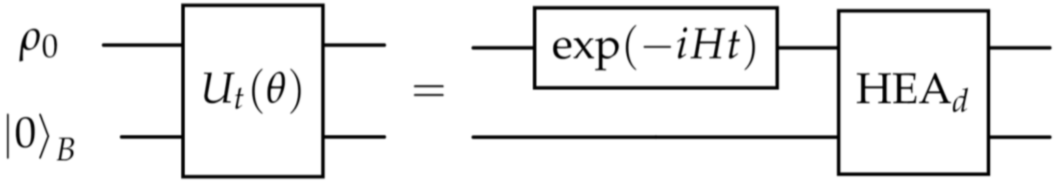
Trotterized specialized hardware-efficient ansatz (trotterized SHEA)

Parametrization of Stinespring unitary

Hardware-efficient ansatz (HEA)



Specialized hardware-efficient ansatz (SHEA)



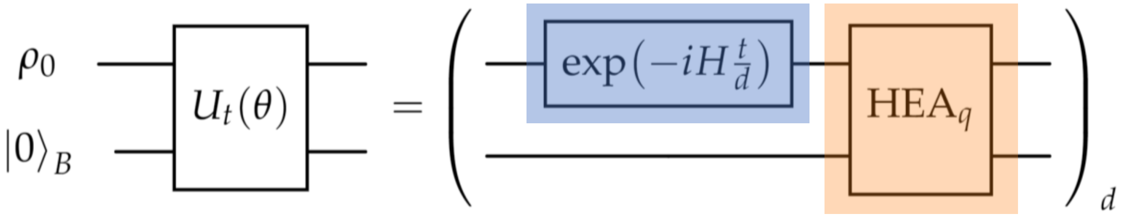
$$\frac{d}{dt}\rho = -\frac{i}{\hbar}[H,\rho] + \sum_k \gamma_k \left(A_k \rho A_k^\dagger - \frac{1}{2}\{A_k^\dagger A_k, \rho\} \right)$$

Interlude: trotterization

For Hermitian operators A and B we have:

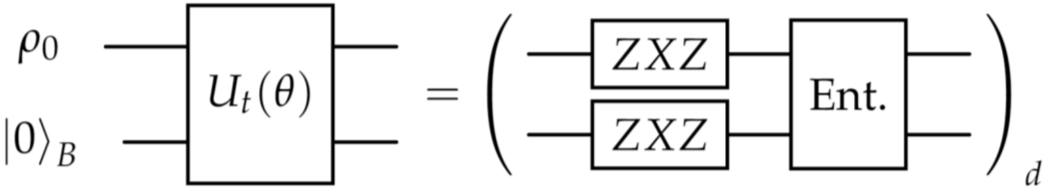
$$\lim_{n\to\infty} \left(e^{-iA\frac{t}{n}} e^{-iB\frac{t}{n}}\right)^n = e^{-i(A+B)t}$$

Trotterized specialized hardware-efficient ansatz (trotterized SHEA)

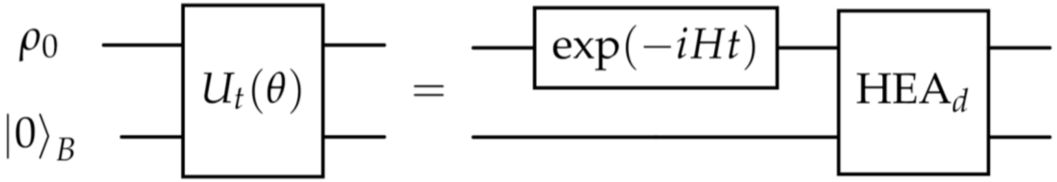


Parametrization of Stinespring unitary

Hardware-efficient ansatz (HEA)



Specialized hardware-efficient ansatz (SHEA)

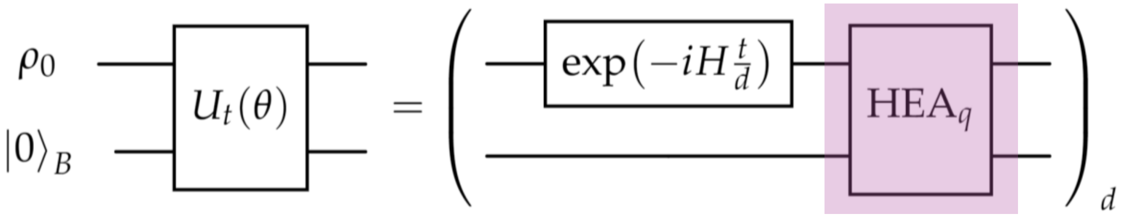


Interlude: trotterization

For Hermitian operators A and B we have:

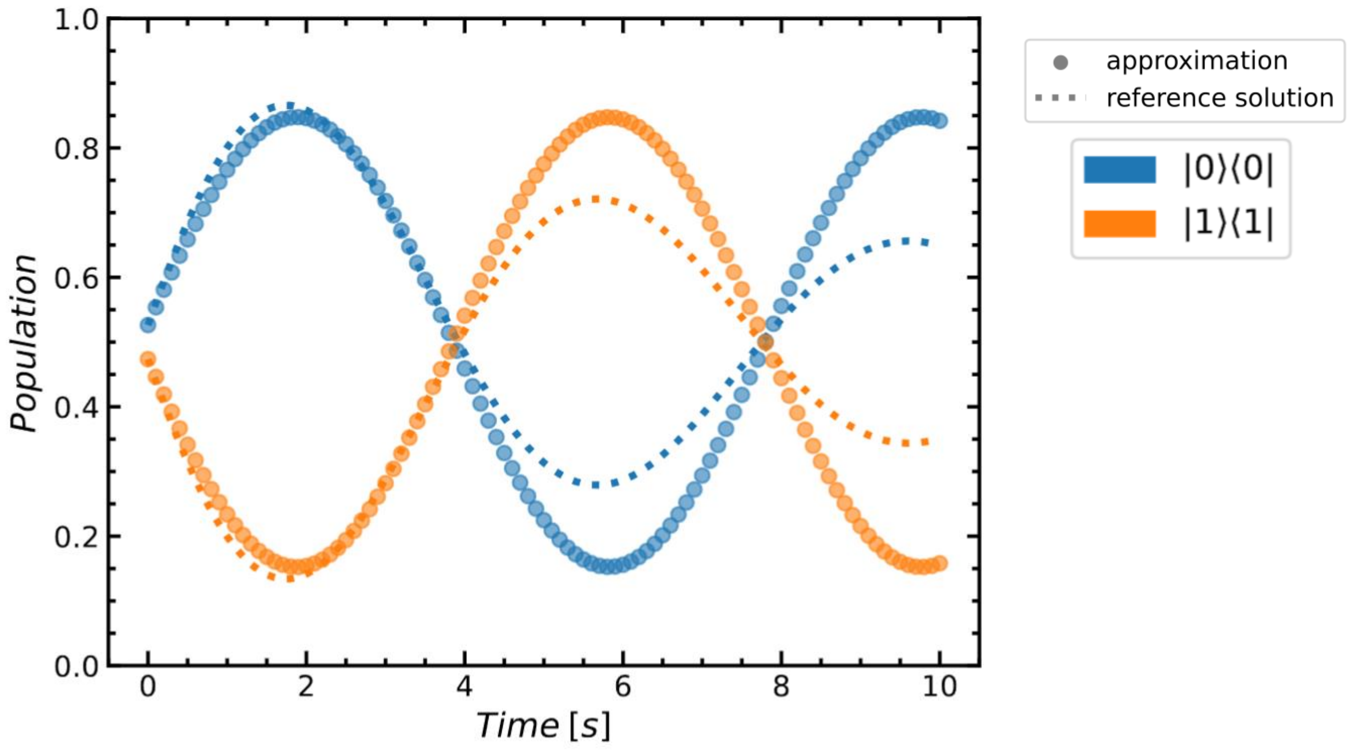
$$\lim_{n\to\infty}\left(e^{-iA\frac{t}{n}}e^{-iB\frac{t}{n}}\right)^n = e^{-i(A+B)t}$$

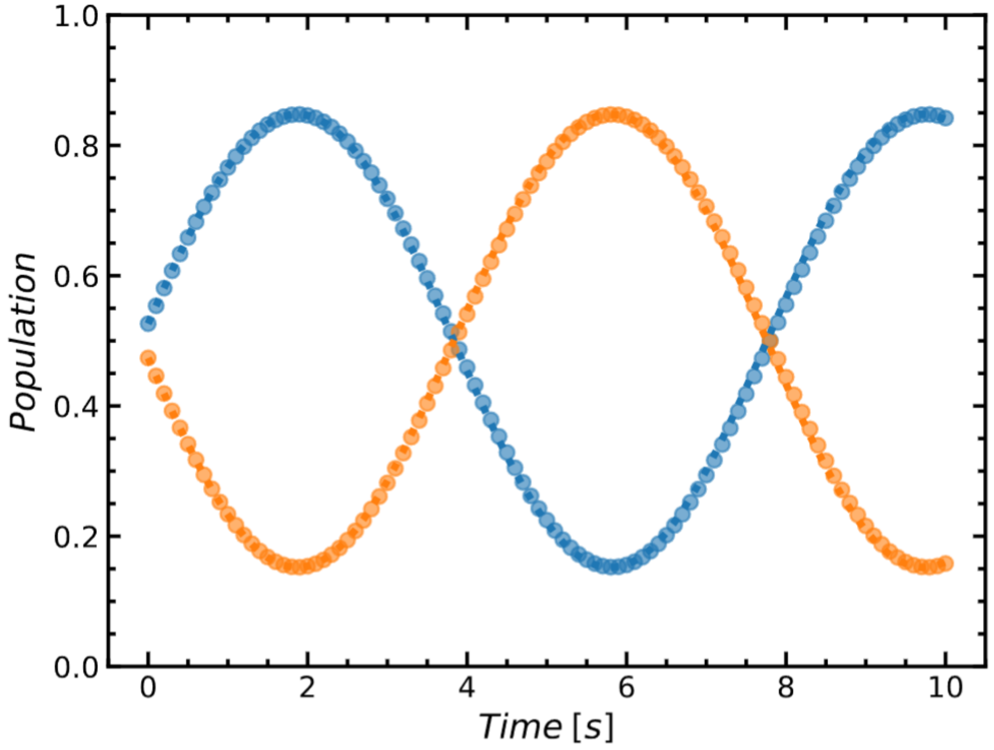
Trotterized specialized hardware-efficient ansatz (trotterized SHEA)

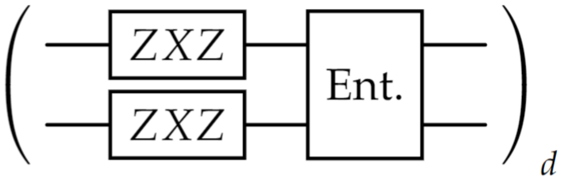


“Same parameters”

Results: approximating Rabi oscillation (one qubit)

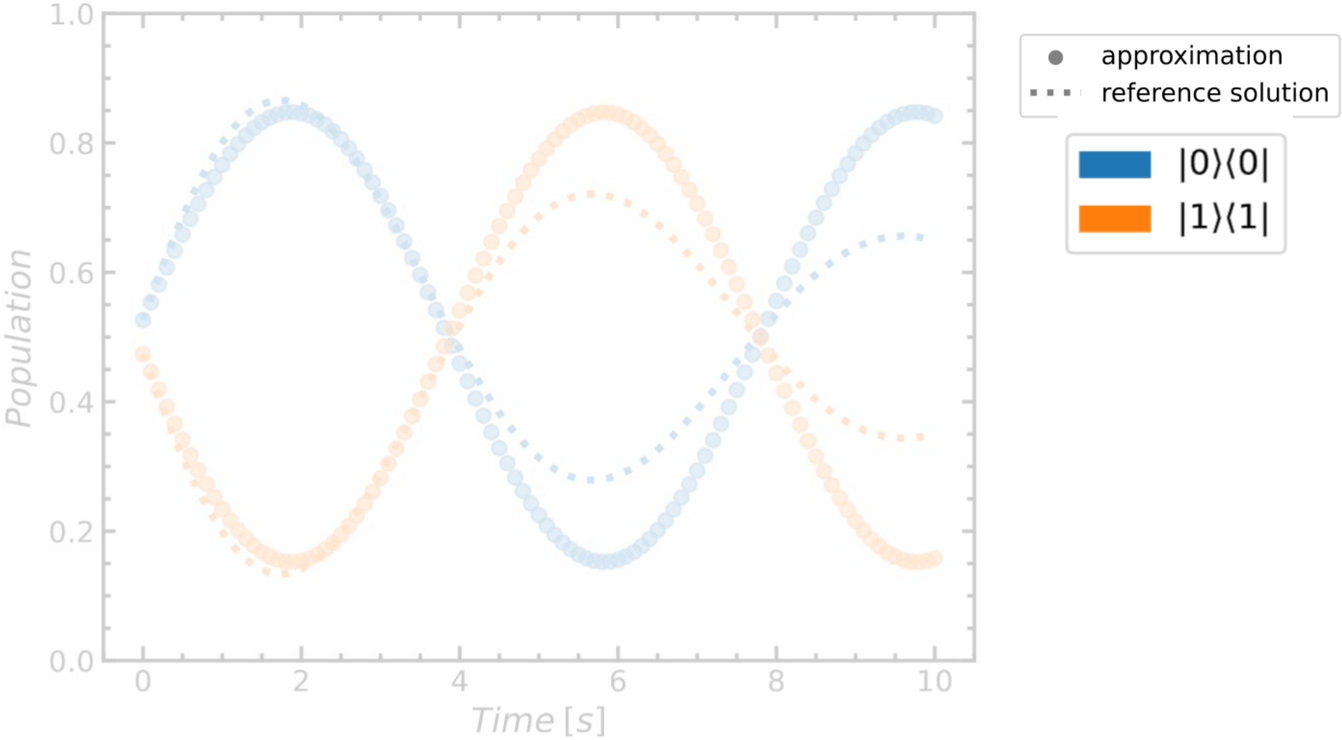


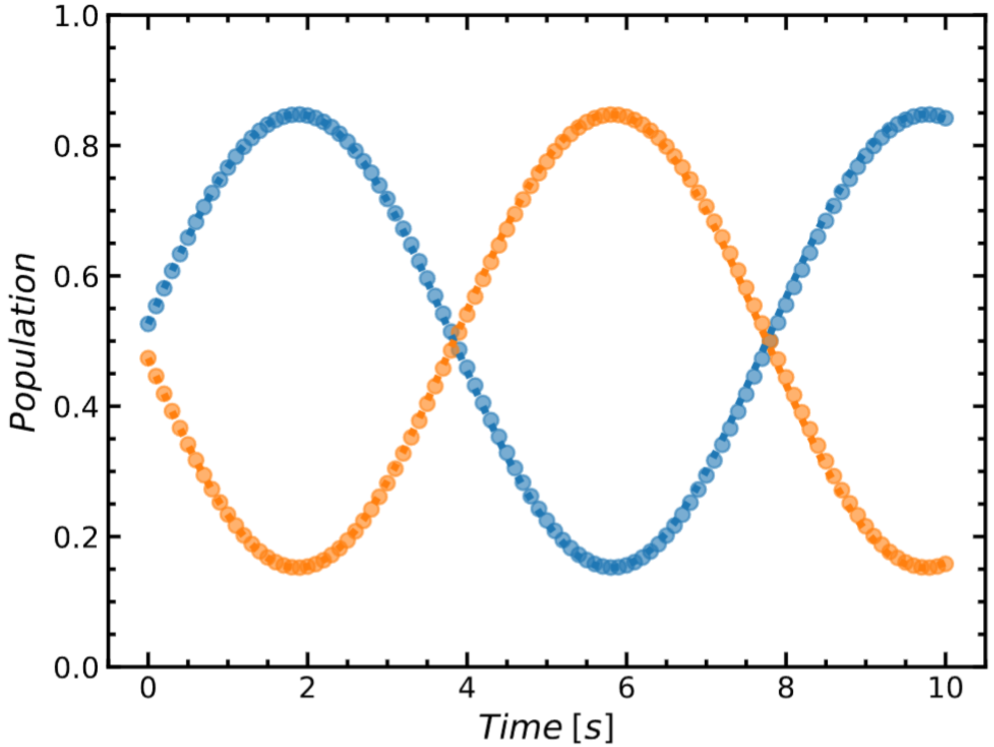


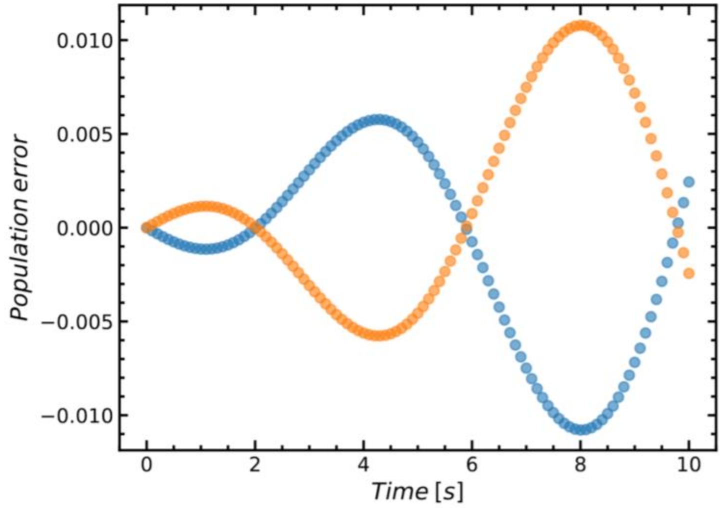


HEA

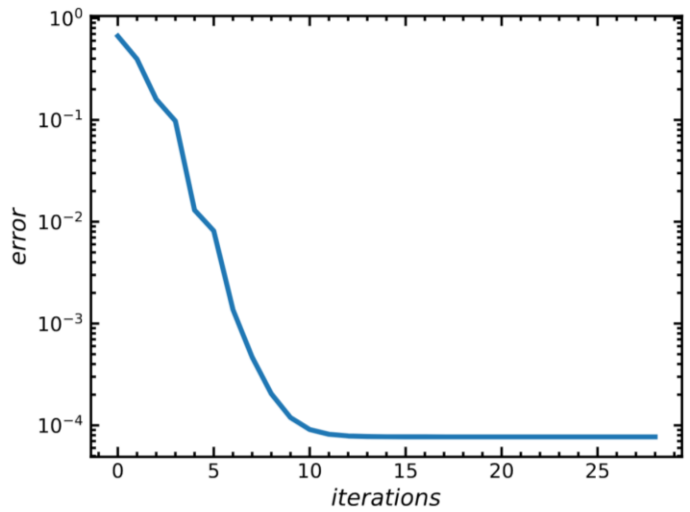
Results: approximating Rabi oscillation (one qubit)



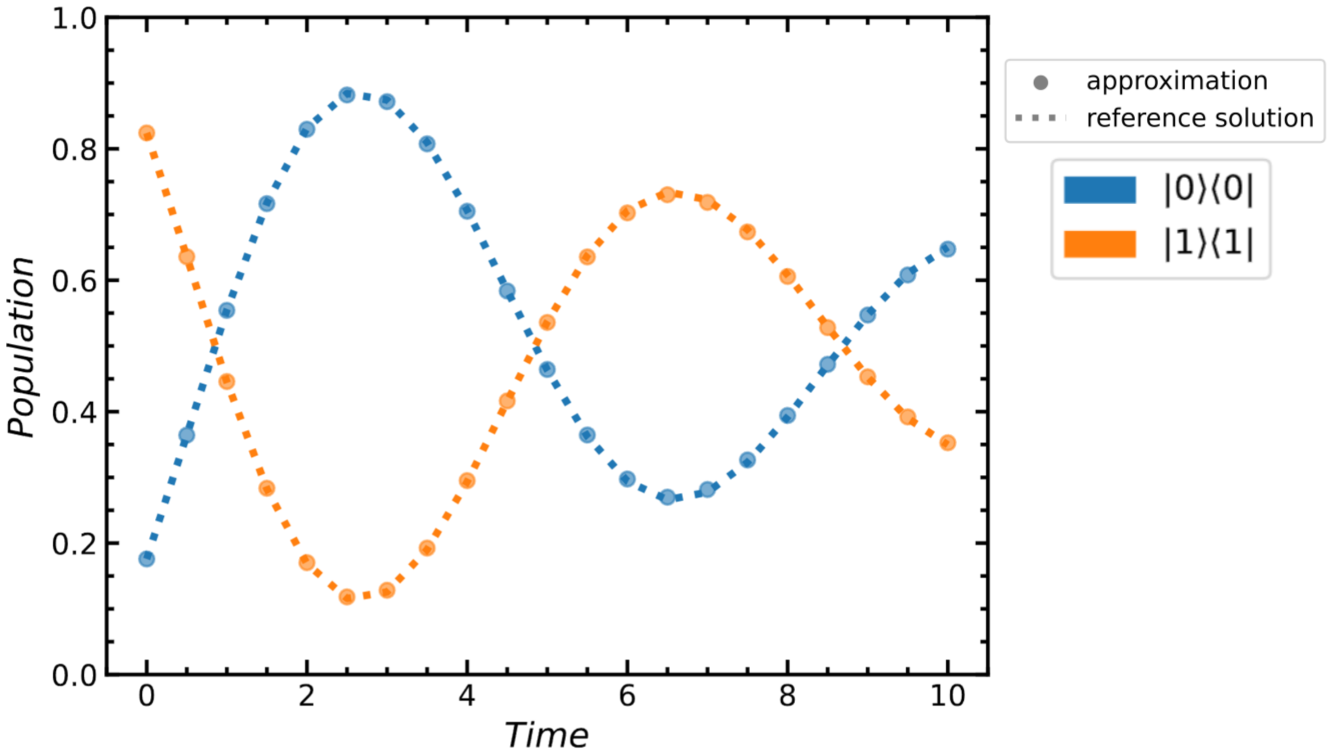


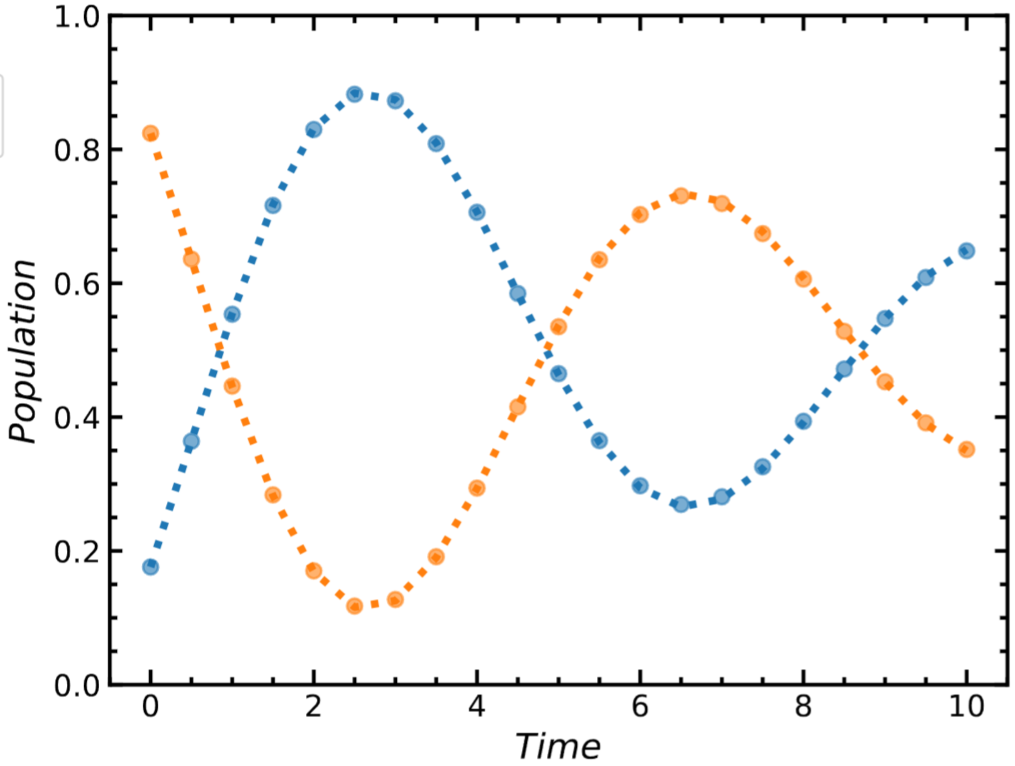


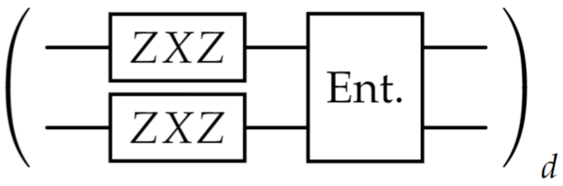
Results: approximating Rabi oscillation (one qubit)



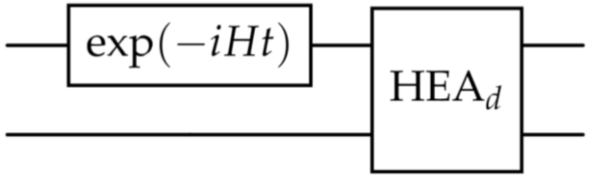
Results: approximating Rabi oscillation (one qubit)





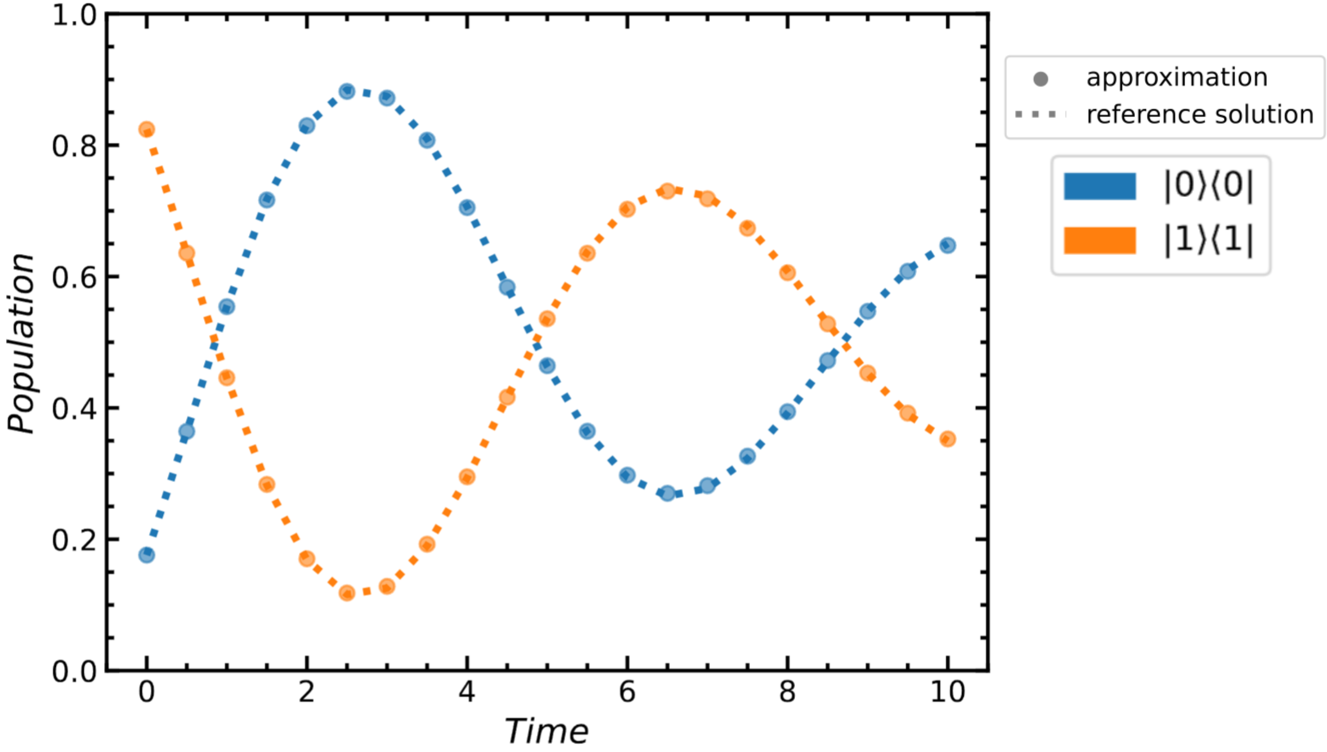


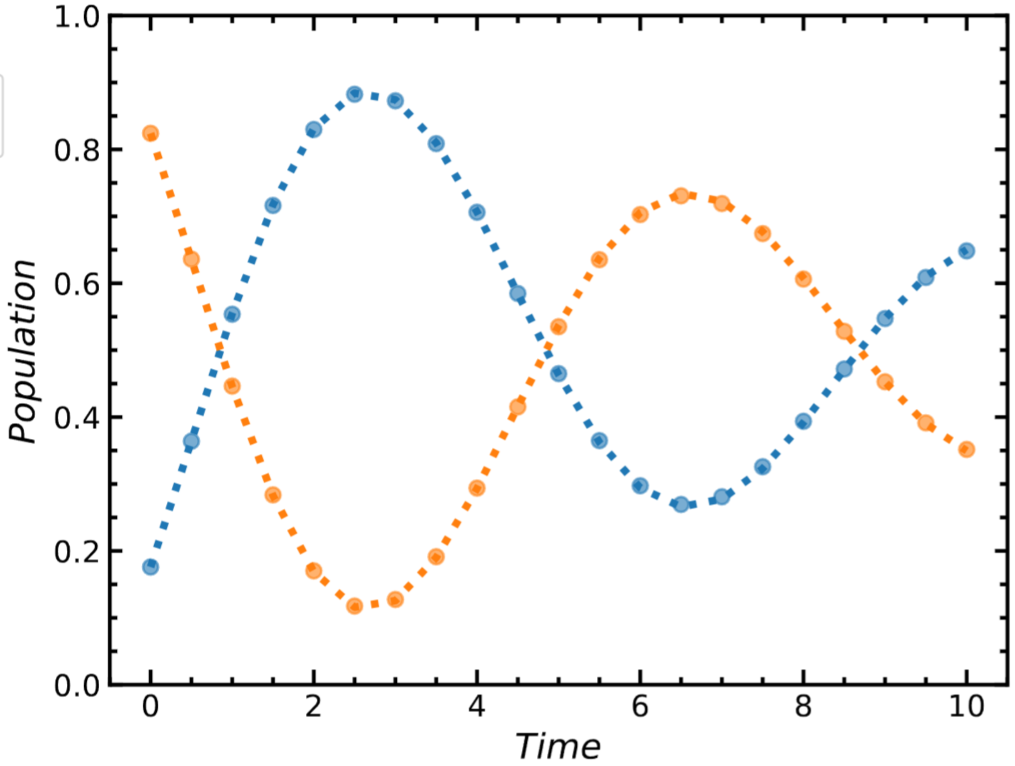
HEA

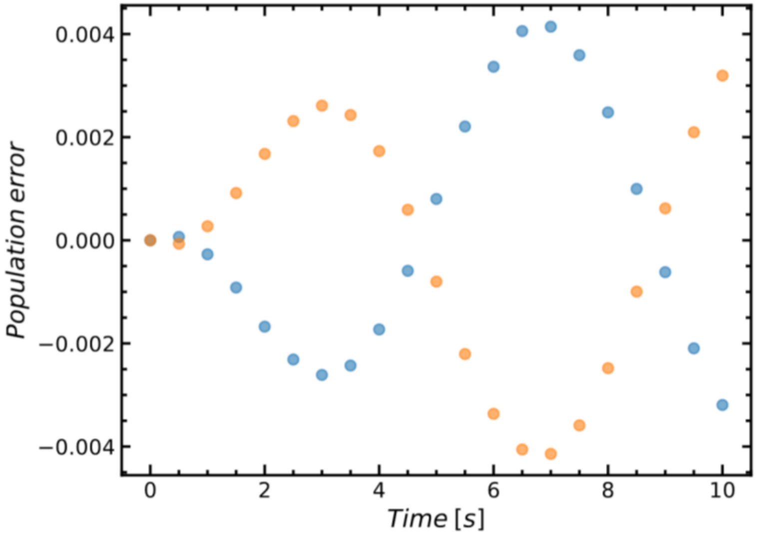


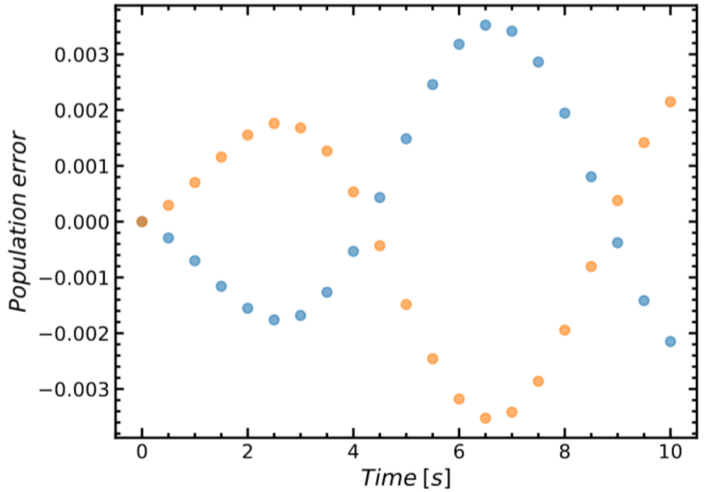
SHEA

Results: approximating Rabi oscillation (one qubit)

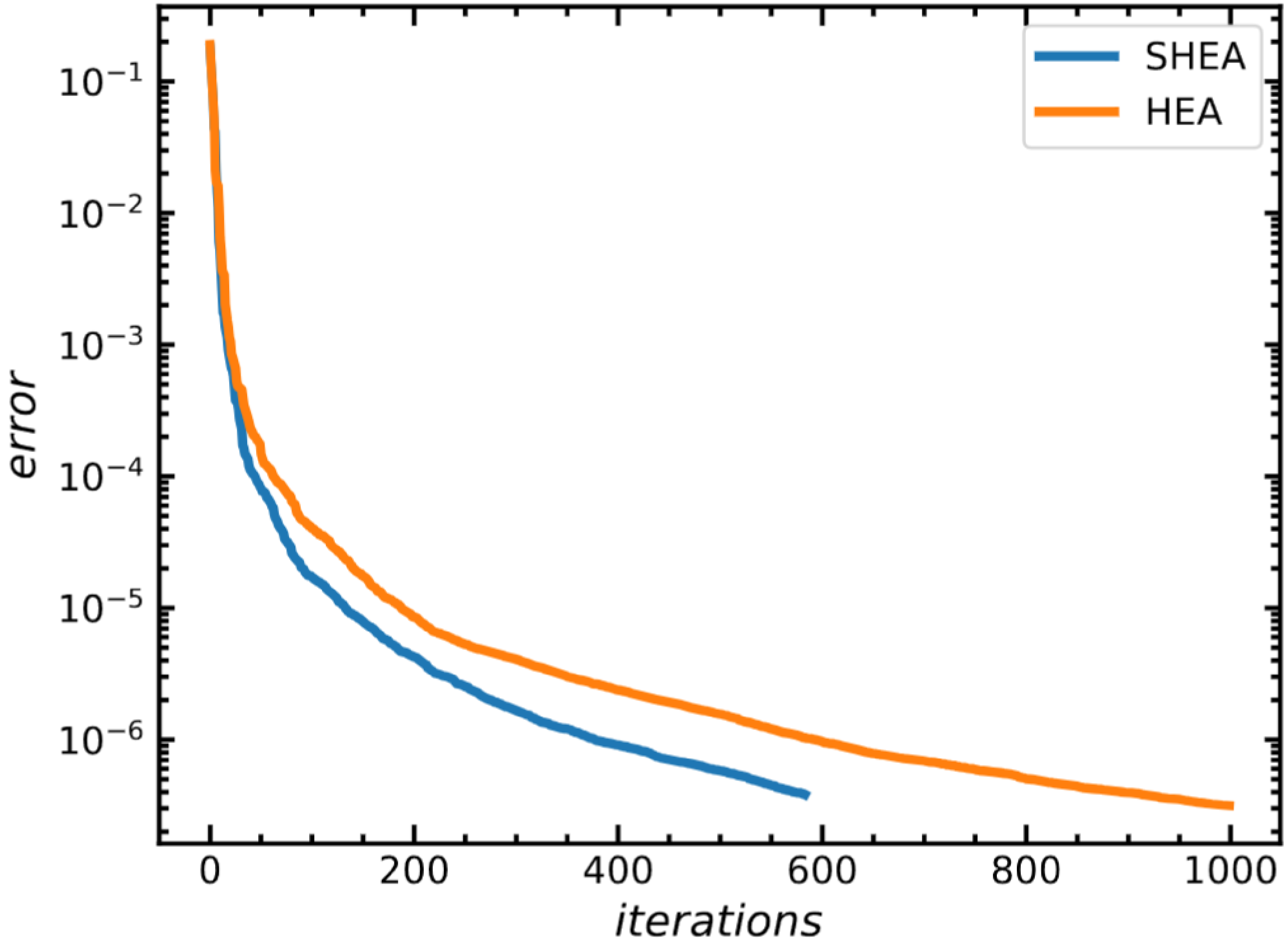


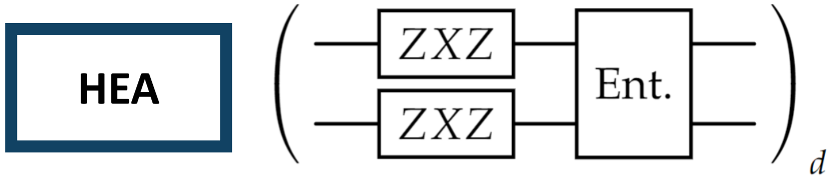


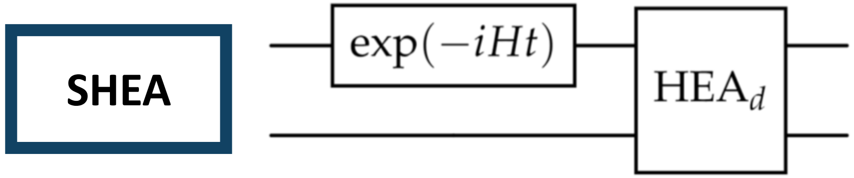




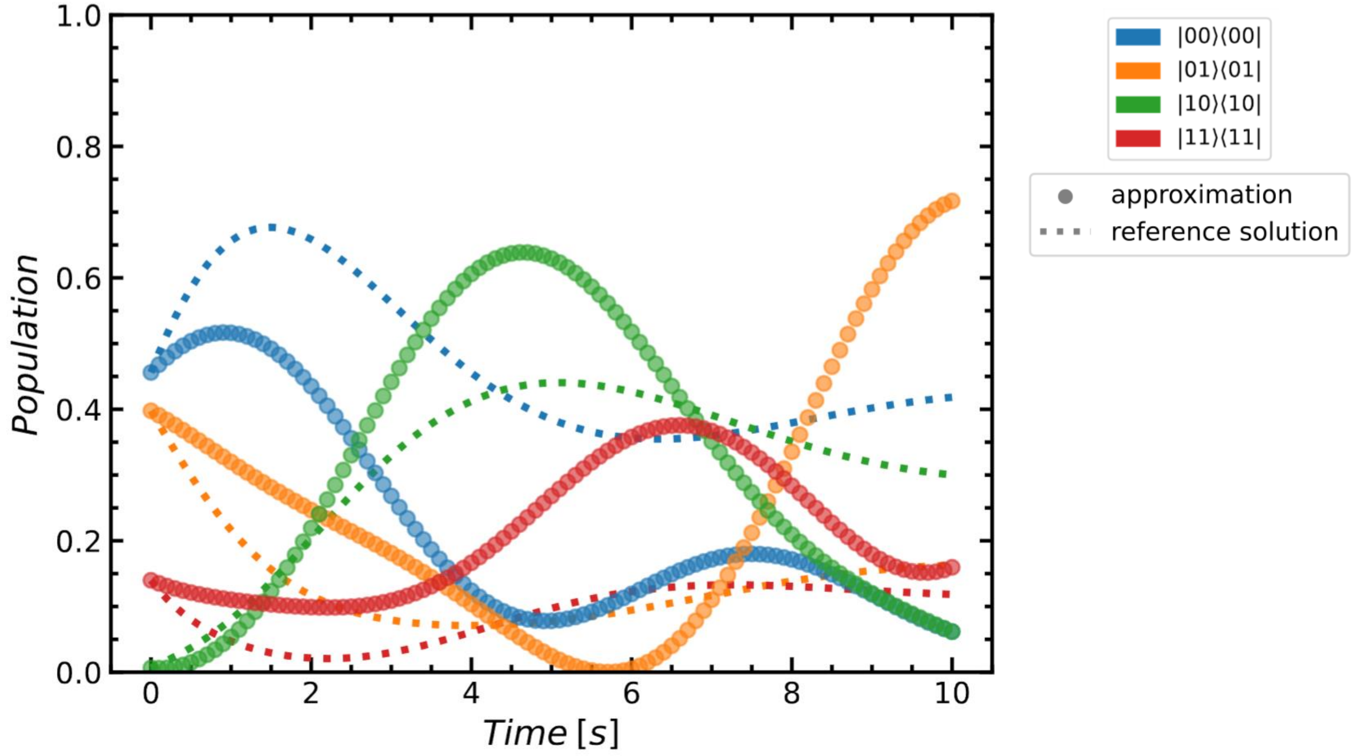
Results: approximating Rabi oscillation (one qubit)

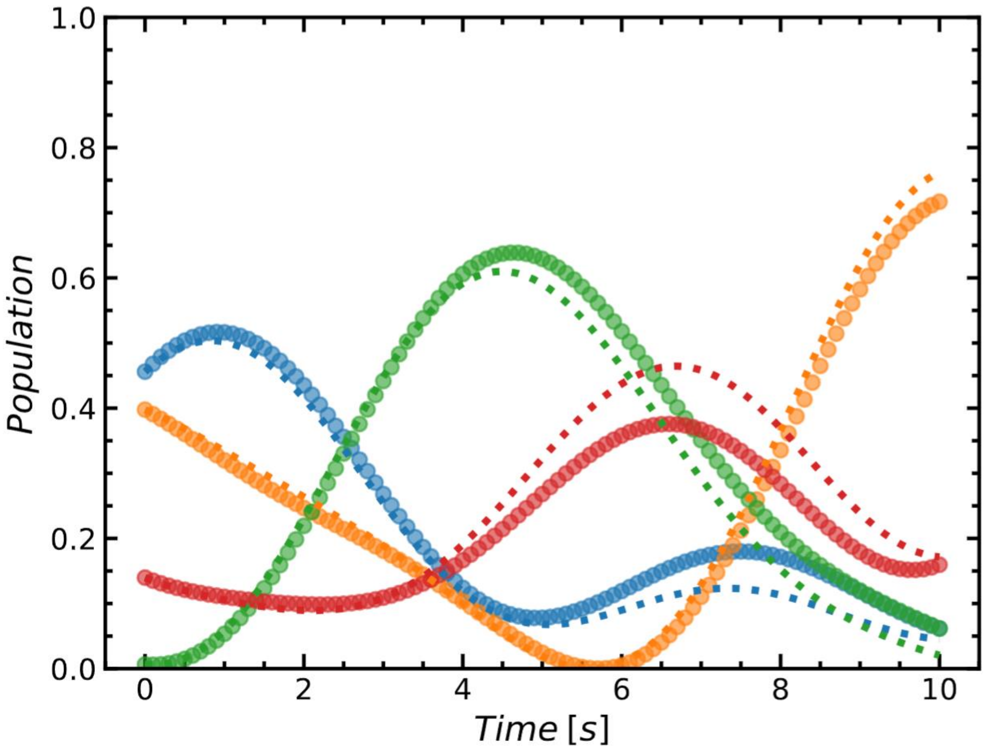


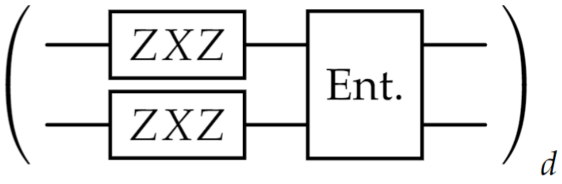




Results: approximating Rabi oscillation (two qubits)

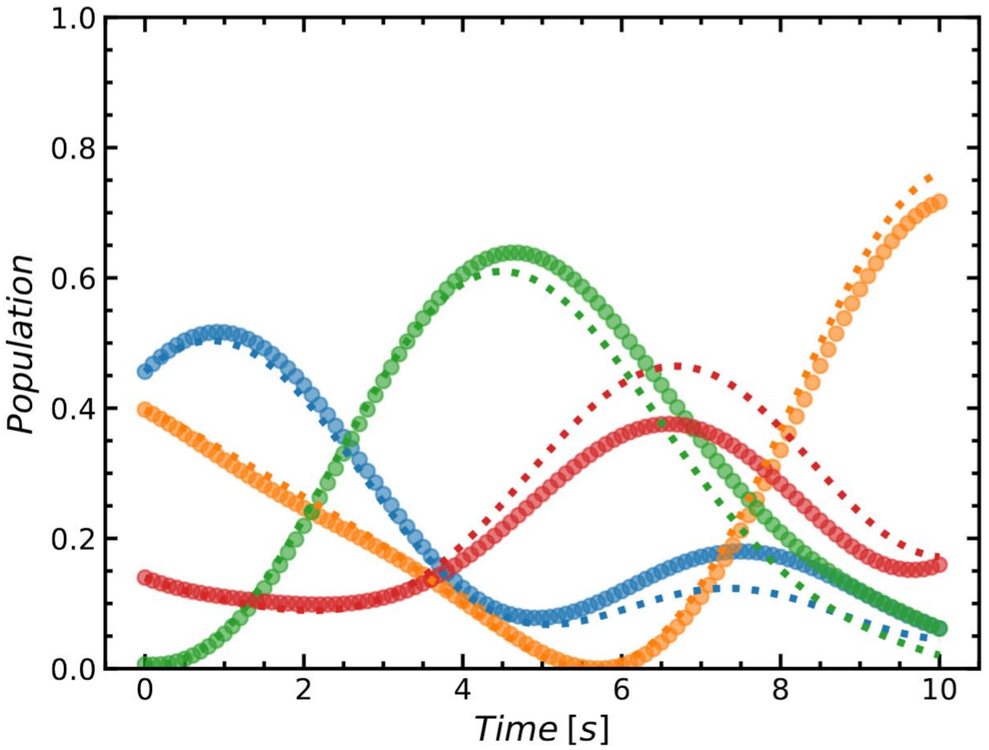


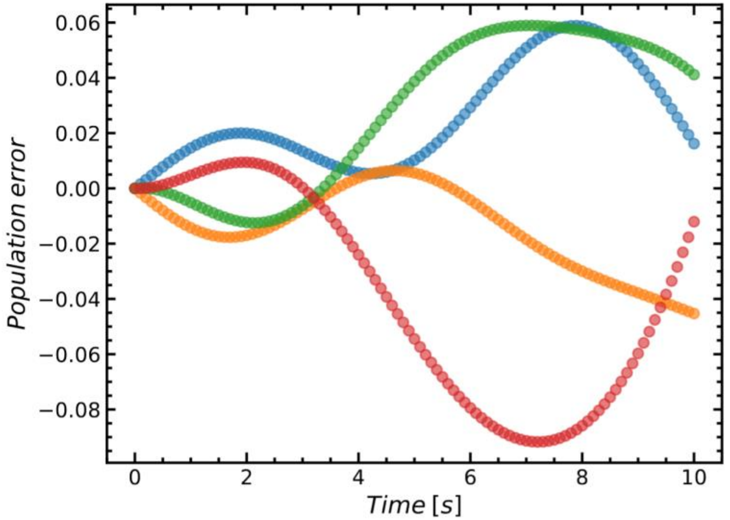




HEA

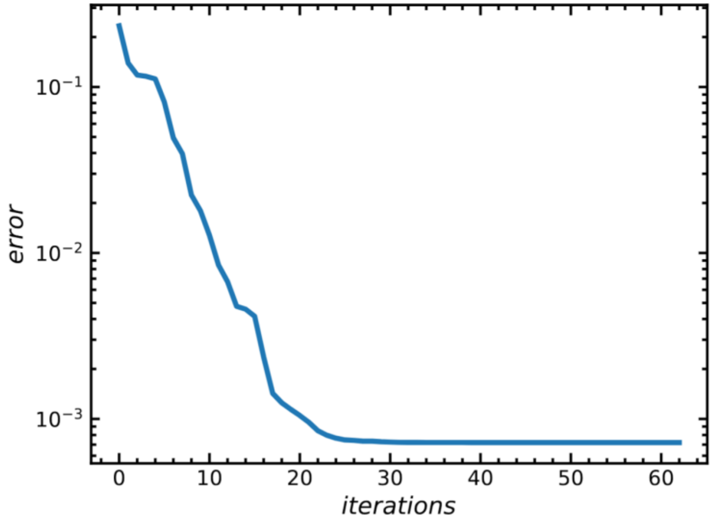
Results: approximating Rabi oscillation (two qubits)



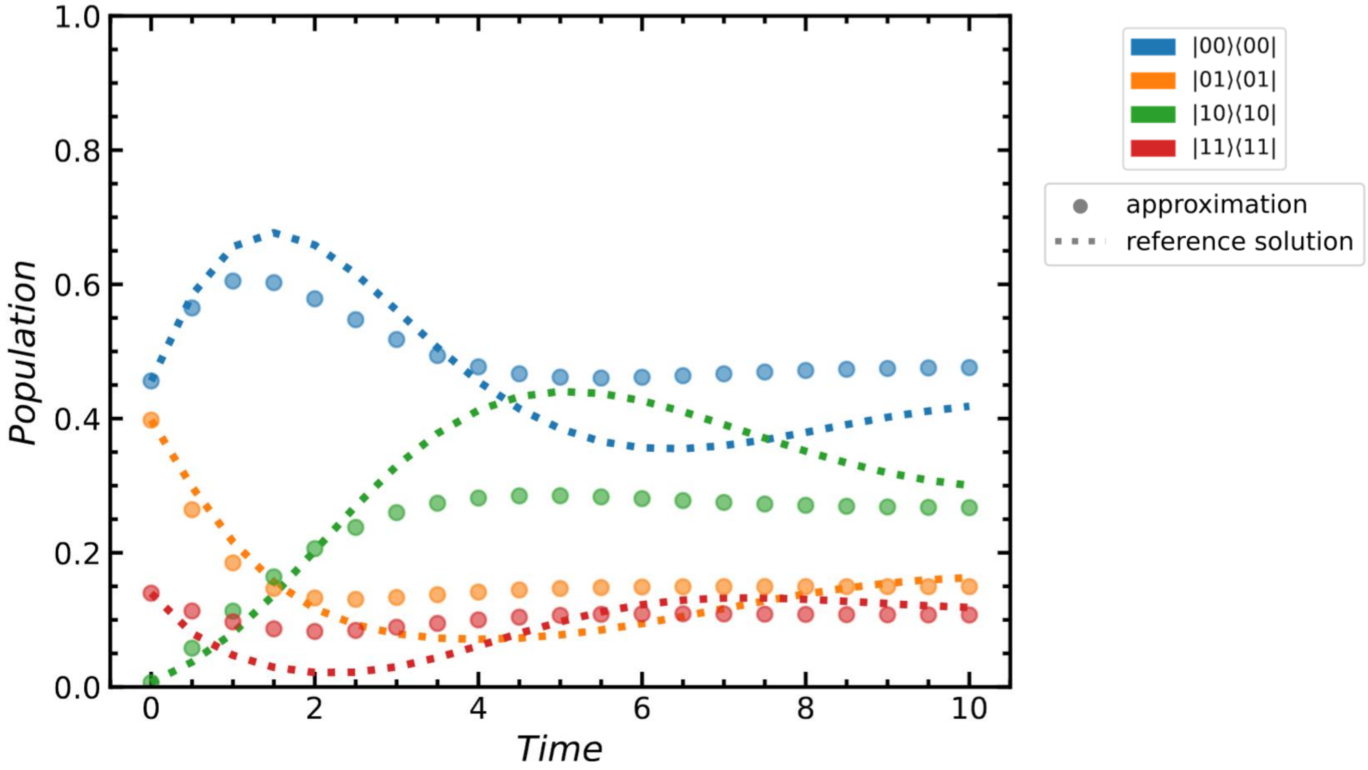


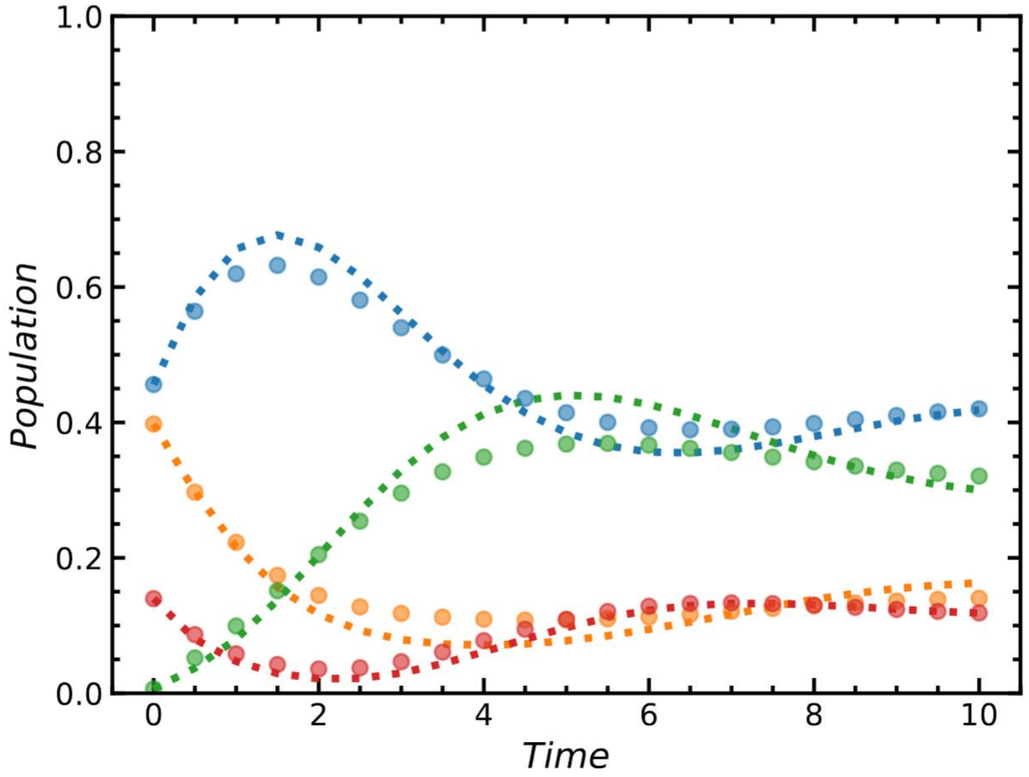
HEA

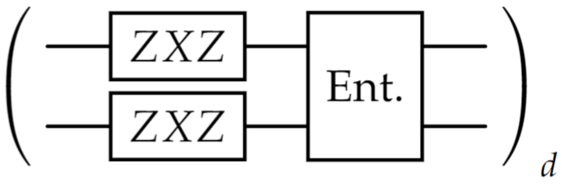
Results: approximating Rabi oscillation (two qubits)



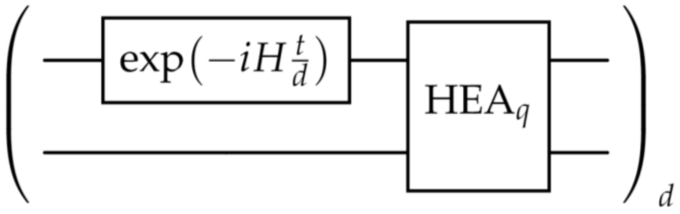
Results: approximating Rabi oscillation (two qubits)





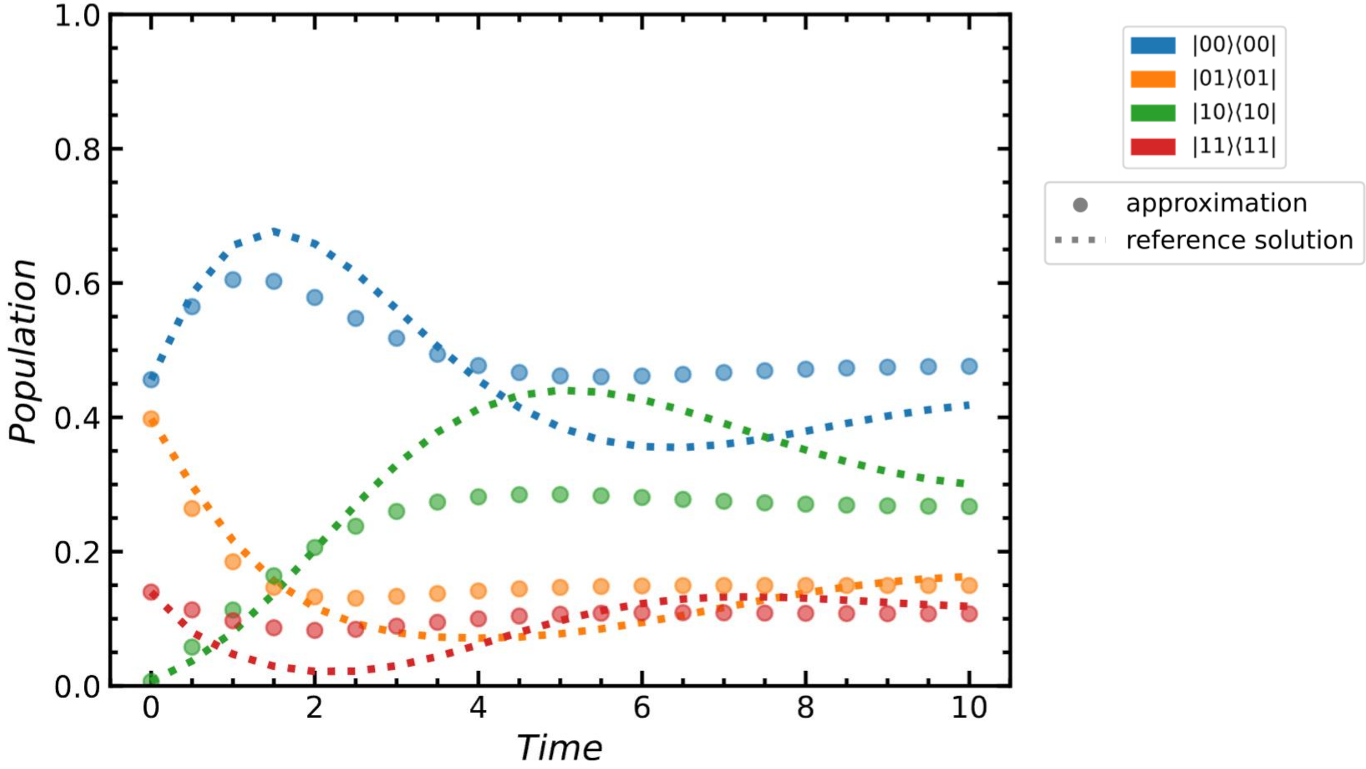


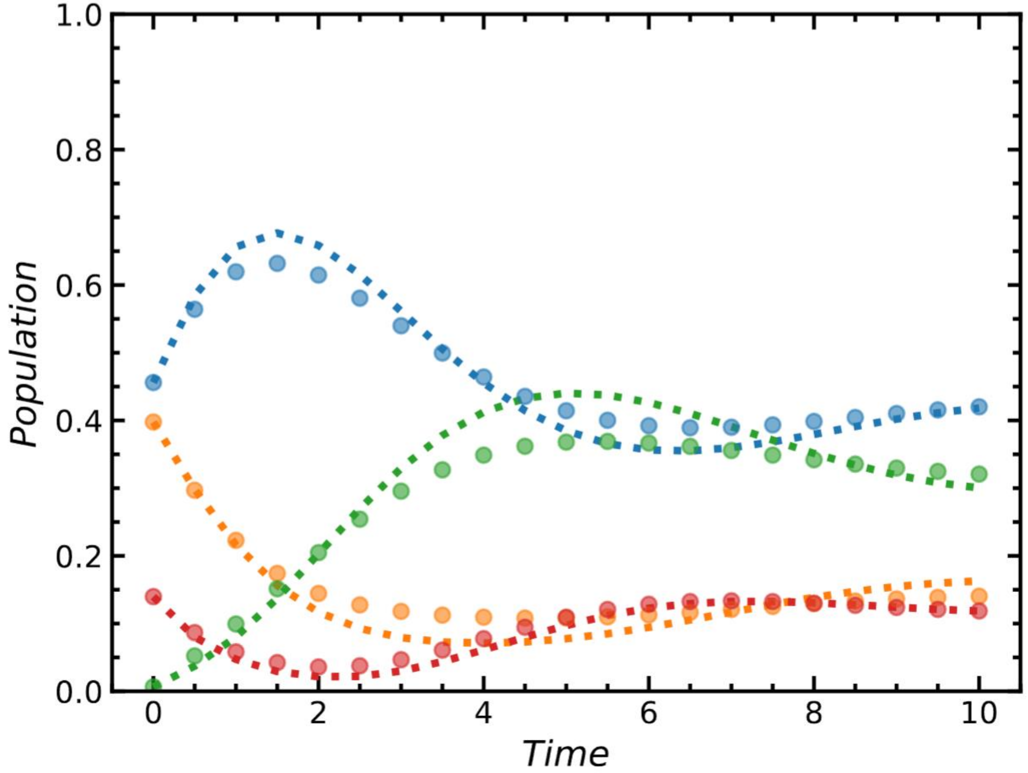
HEA

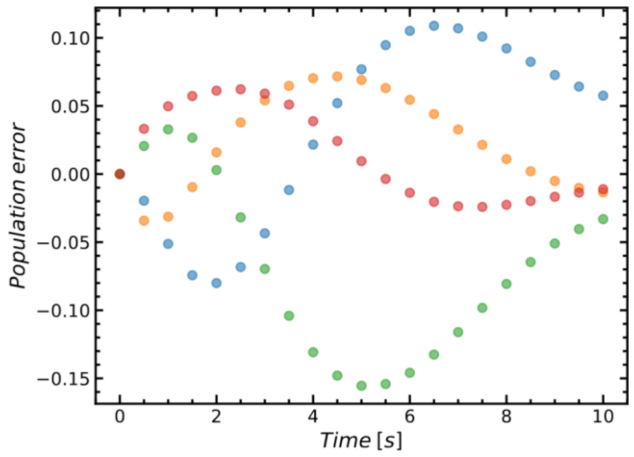


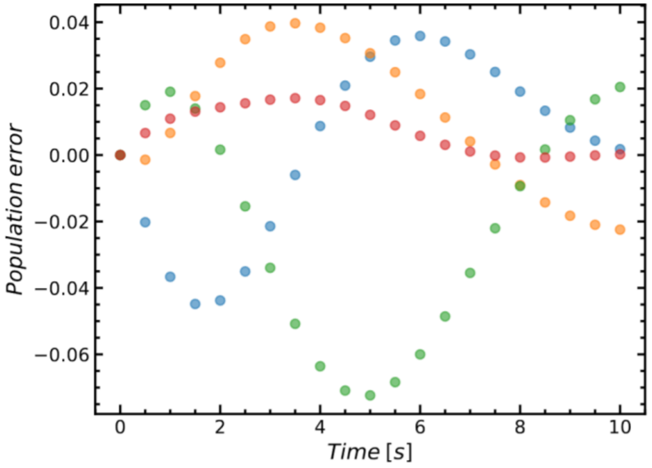
Trotterized SHEA

Results: approximating Rabi oscillation (two qubits)

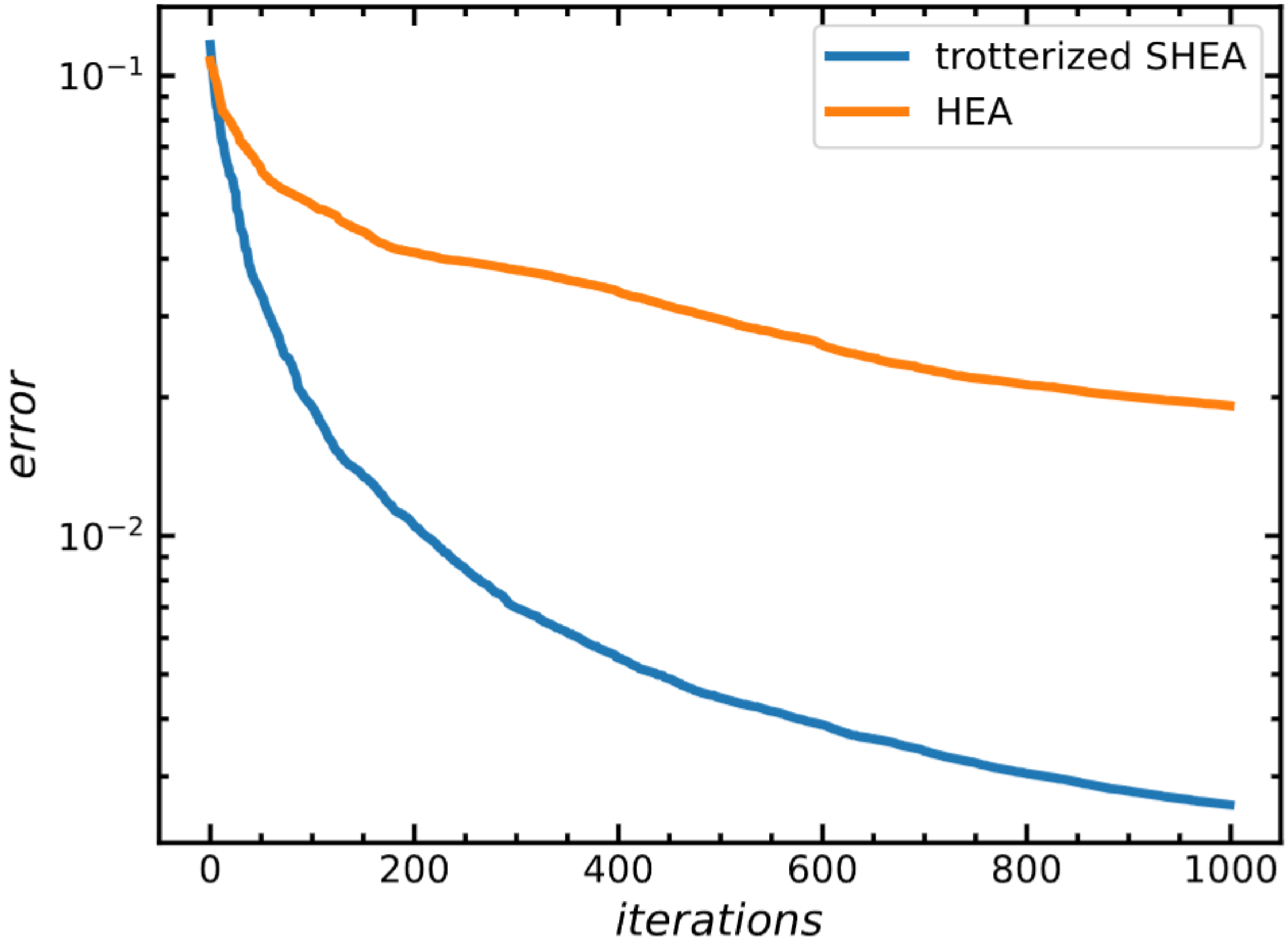


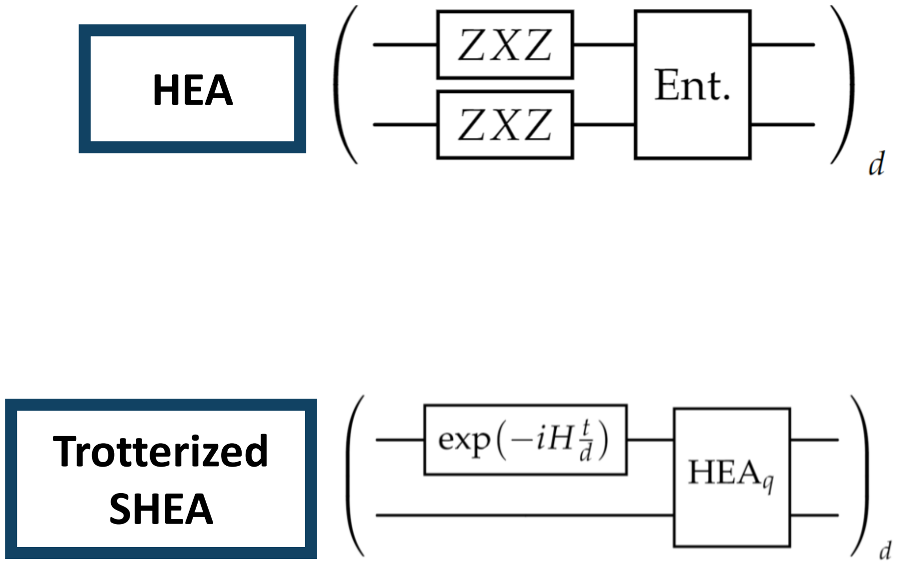






Results: approximating Rabi oscillation (two qubits)





Future research

- **How much training data is needed?**
- **What is the relation between depth and expressibility?**

- **Implement on quantum computer**