From Easy to Hard: Tackling Quantum Problems with Learned Gadgets

Akash Kundu (University of Helsinki), Leopoldo Sarra (Flatiron Institute)

Motivation

Optimization of Quantum Circuits

- Find a quantum circuit that solves a given task
- Learn by solving a parametrized set of problems with various difficulty

Method

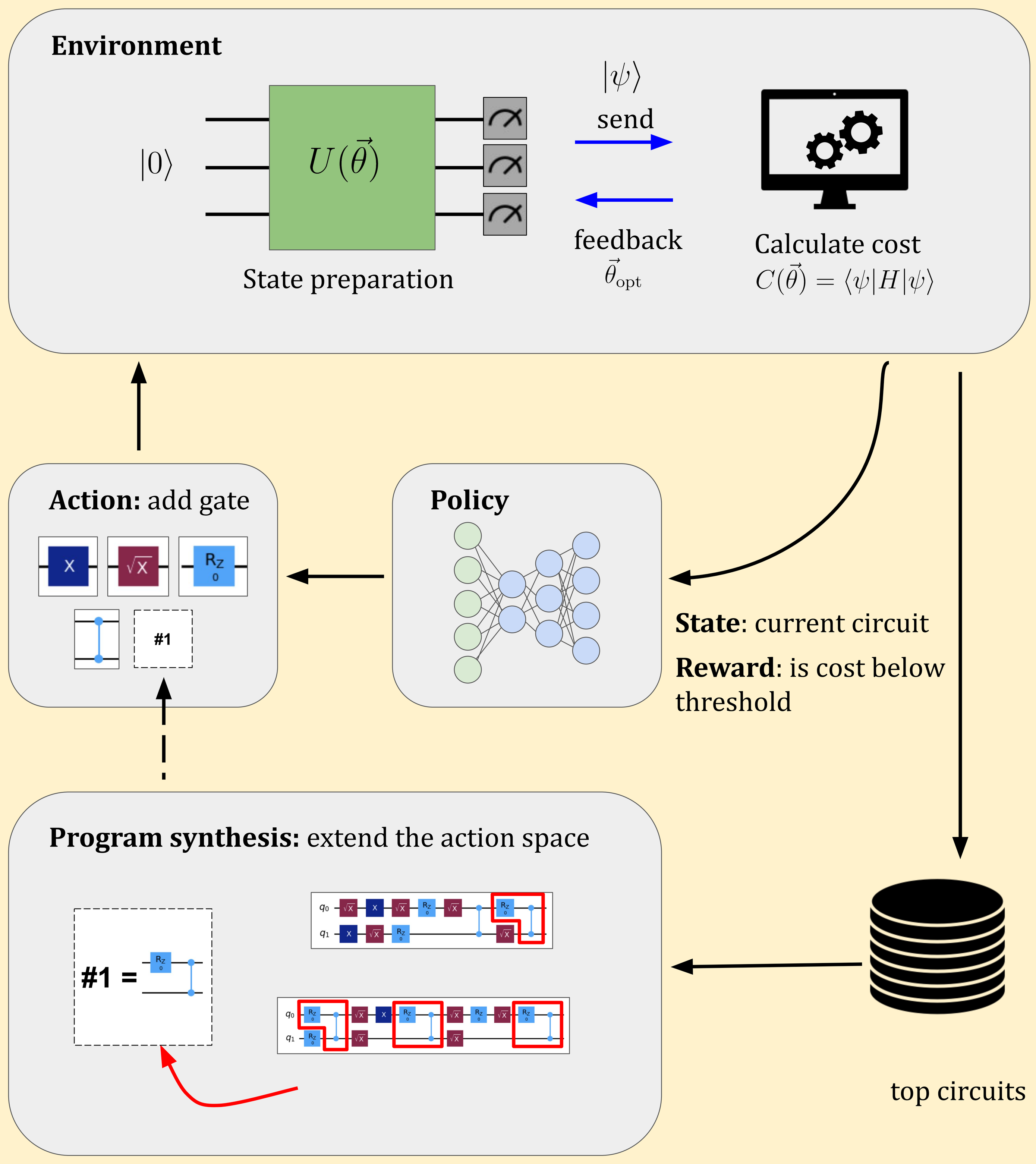
An iterative algorithm

1) Reinforcement learning

- Propose a circuit for state preparation
- Minimize state energy

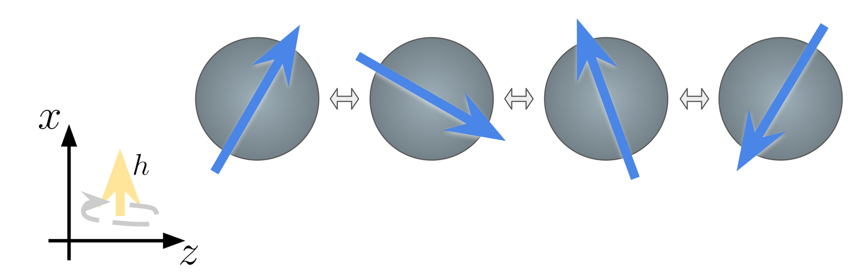
2) Program synthesis

- Analyze best proposed circuits
- Extract the most useful composite gate
- Expand the reinforcement learning agent's action space



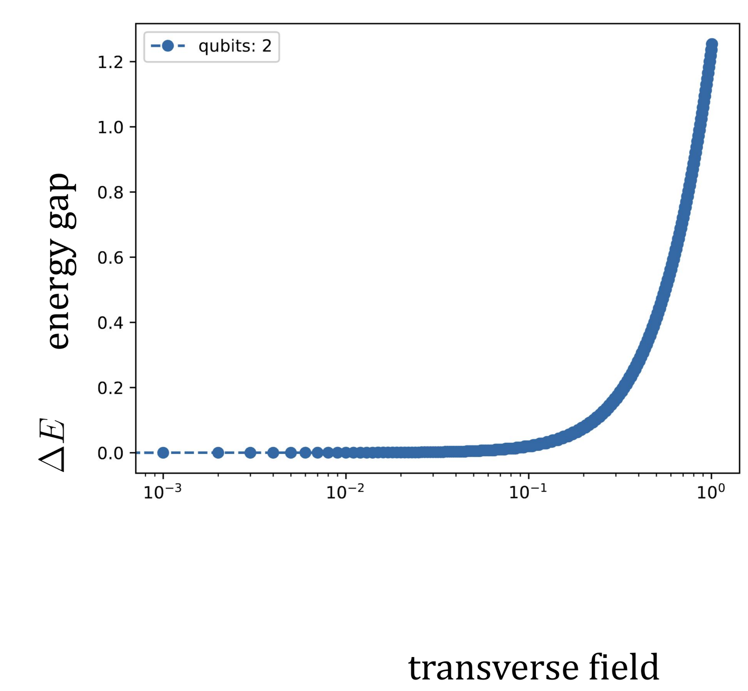
Application

The Transverse-Field Ising Model



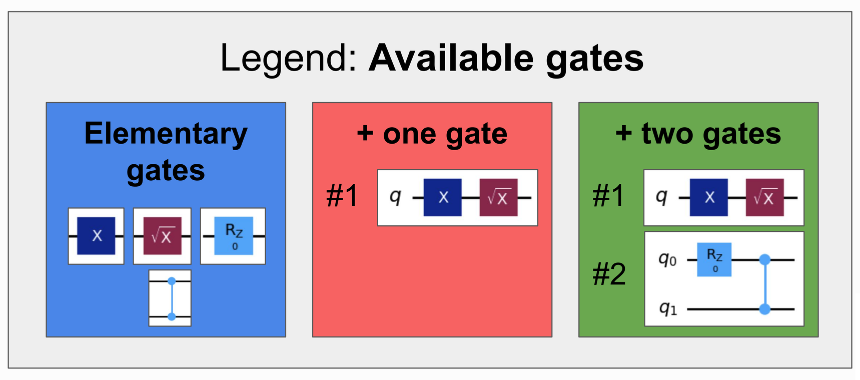
$$H = -J \sum_{\langle i,j \rangle}^{N} \sigma_i^z \sigma_j^z - h \sum_i \sigma_i^x$$

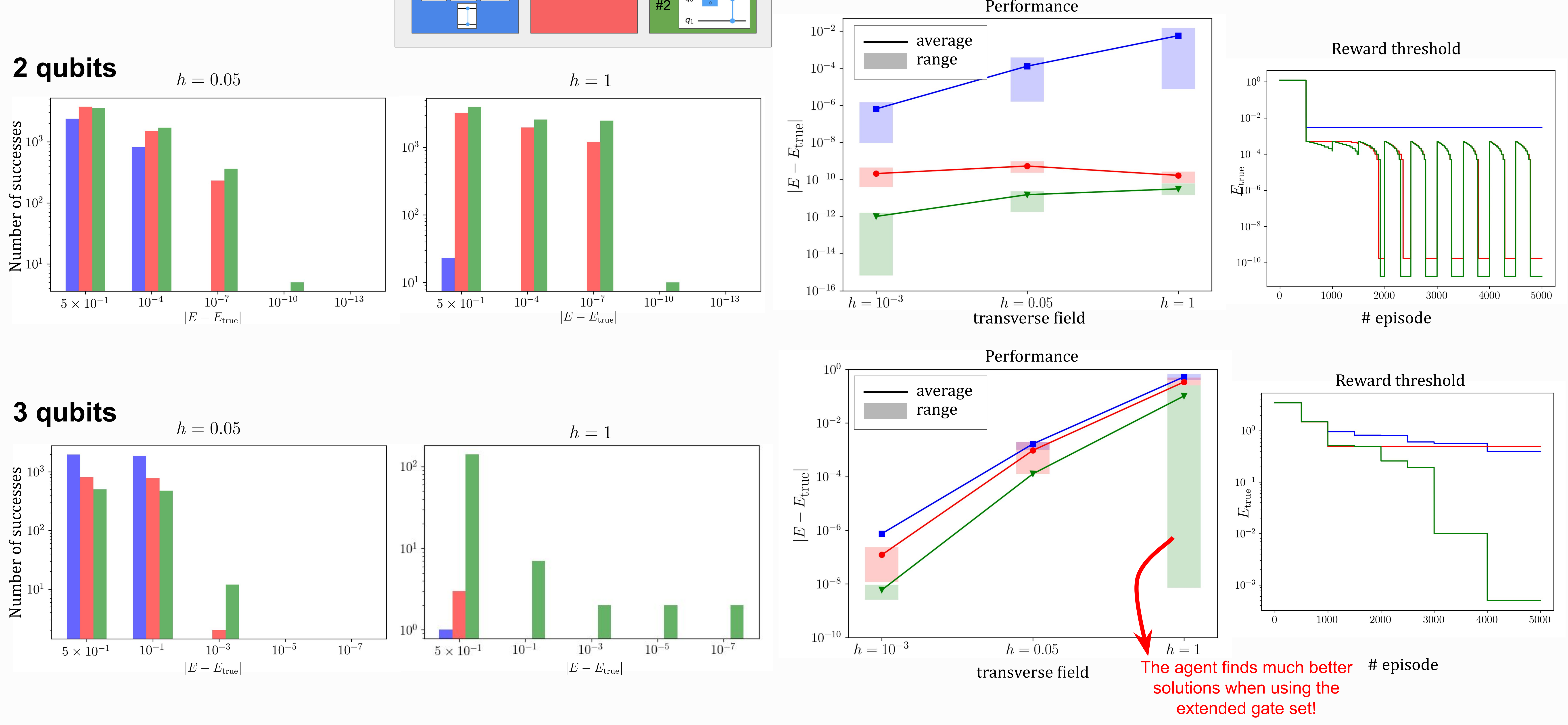
GOAL: build a circuit that prepares the ground state of the system



The strength of the external field tunes the difficulty of the problem

Results





Outlook

- Improve reinforcement learning efficiency by extending action space
- Generalization to harder regimes and larger number of qubits
- Can be directly optimized for specific real hardware

References

- A. Kundu, L. Sarra, From Easy to Hard: Tackling Quantum Problems with Learned Gadgets On Real Hardware, arXiv: arXiv:2411.00230
- Y. J. Patel, A. Kundu et al., Curriculum reinforcement learning for quantum architecture search under hardware errors, arXiv:2402.03500
- A. Kundu, P. Bedełek et al., Enhancing variational quantum state diagonalization using reinforcement learning techniques, New J. Phys. 26 013034
- L.Sarra, K. Ellis, F. Marquardt, Discovering Quantum Circuit components with Program Synthesis, MLST 5 (2) 025029

HELSINGIN YLIOPISTO HELSINGFORS UNIVERSITET UNIVERSITY OF HELSINKI

SIMONS FOUNDATION

Machine Learning for Quantum Technologies Workshop
Max Planck Institute for the Science of Light
5 - 8 November 2024, Erlangen Germany

MAX-PLANCK-INSTITUT FÜR DIE PHYSIK DES LICHTS

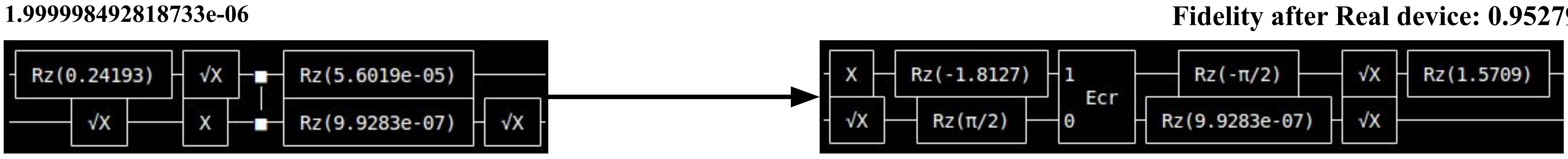
Zoom Link: https://helsinki.zoom.us/j/65033257478?pwd=XFvcXVCPVURfeSckFj5NITBbc8pZqq.1

References

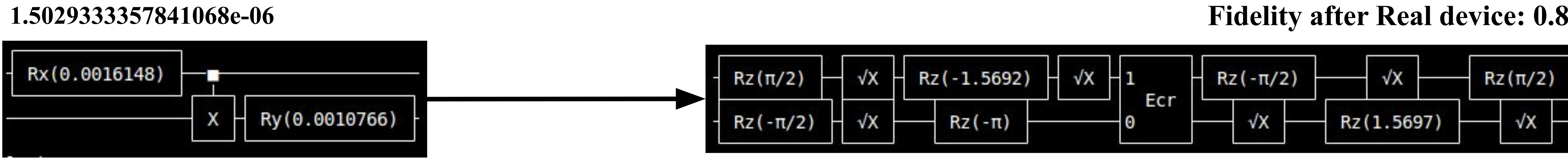
1. https://arxiv.org/abs/2306.11086
2. https://arxiv.org/abs/2402.03500

Real device run

1.999998492818733e-06 **Fidelity after Real device: 0.9527998229651132**



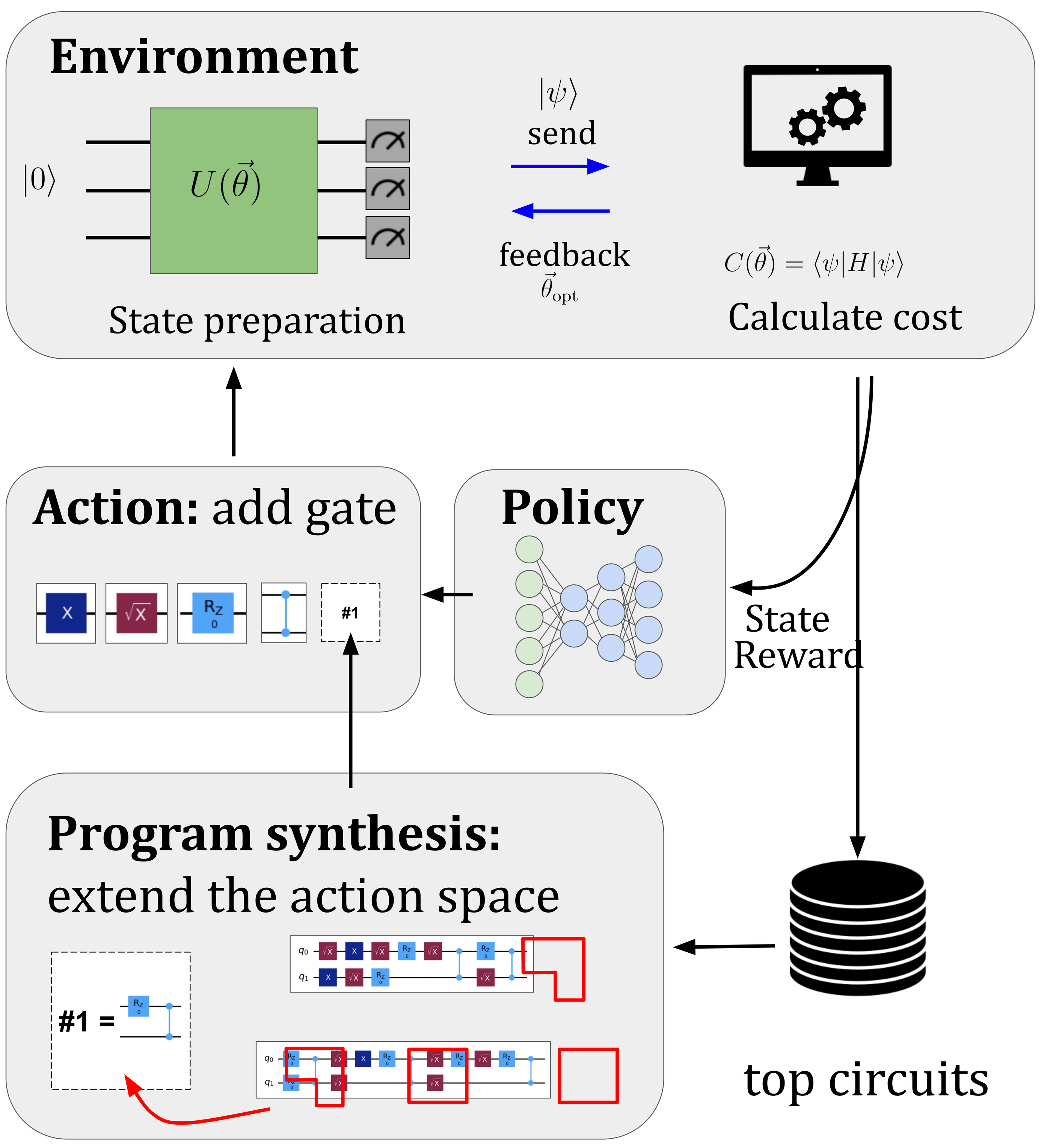
1.5029333357841068e-06 **Fidelity after Real device: 0.8782**



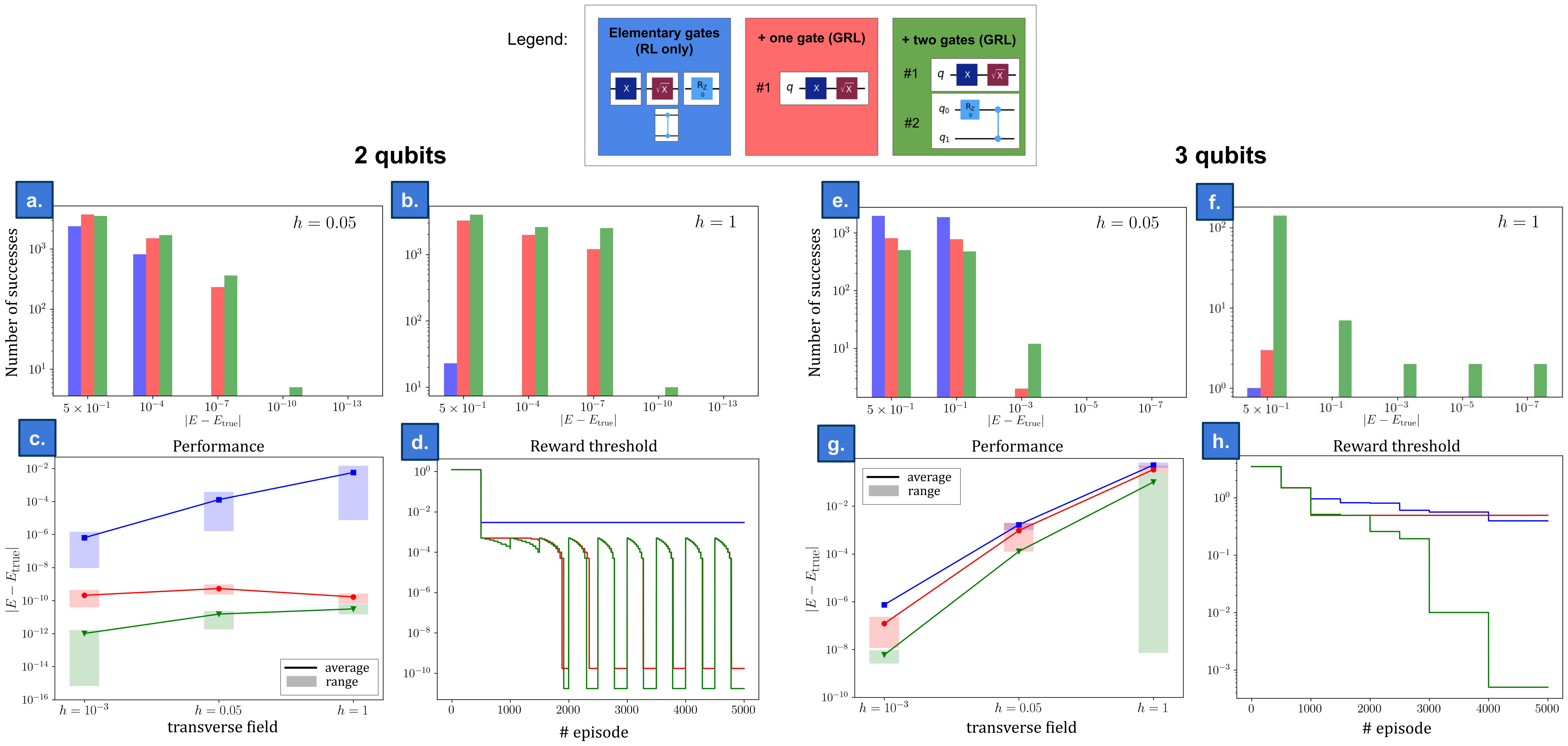
ROT VARIABLE
REWARD

Max 20, min 10

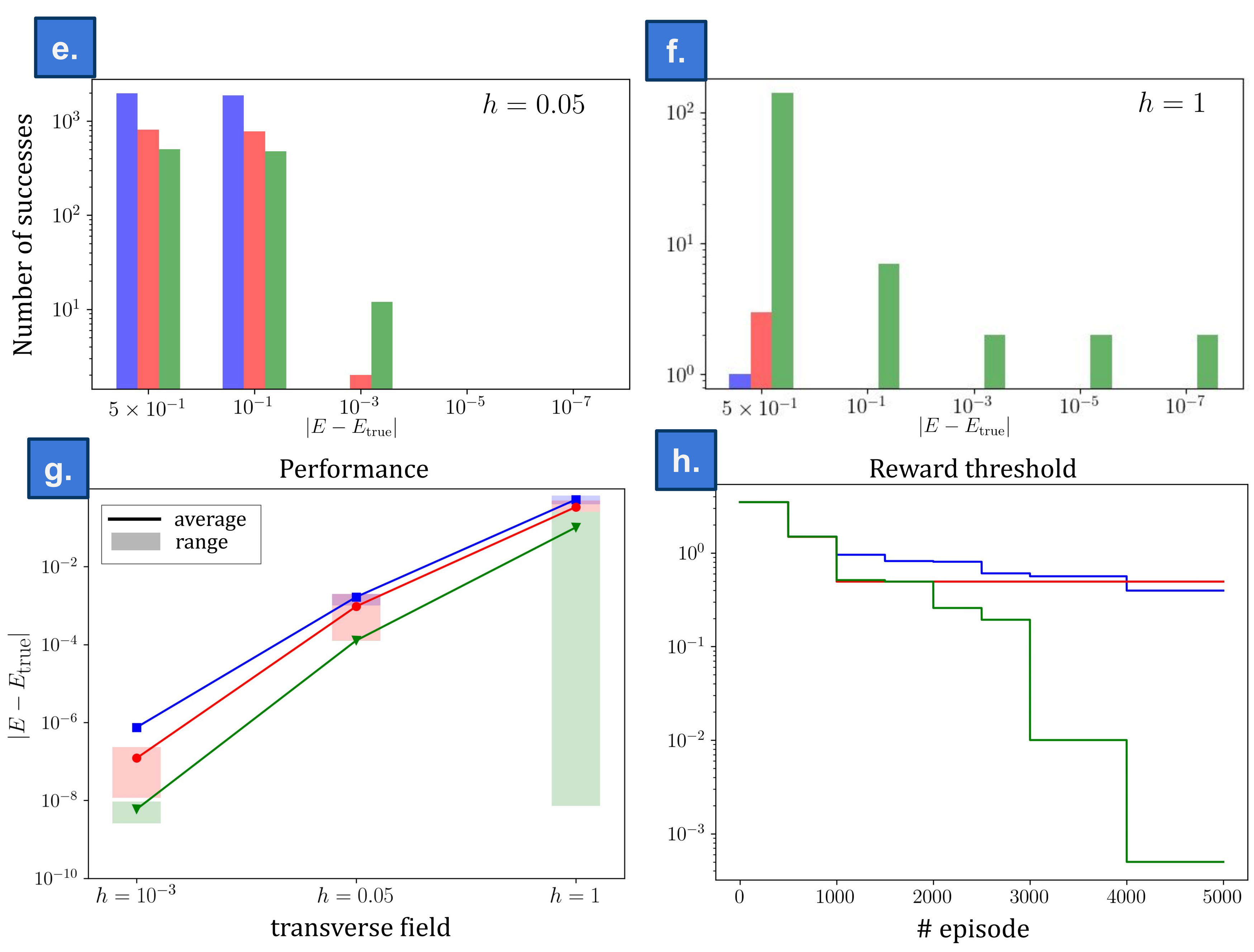
R = r*-log(1/numb of rot+1)

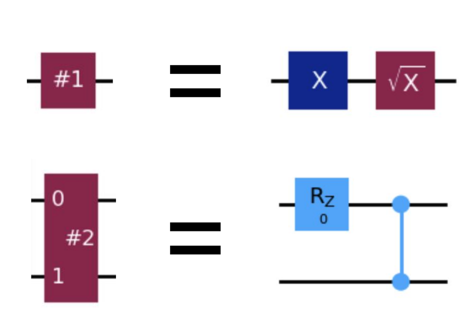


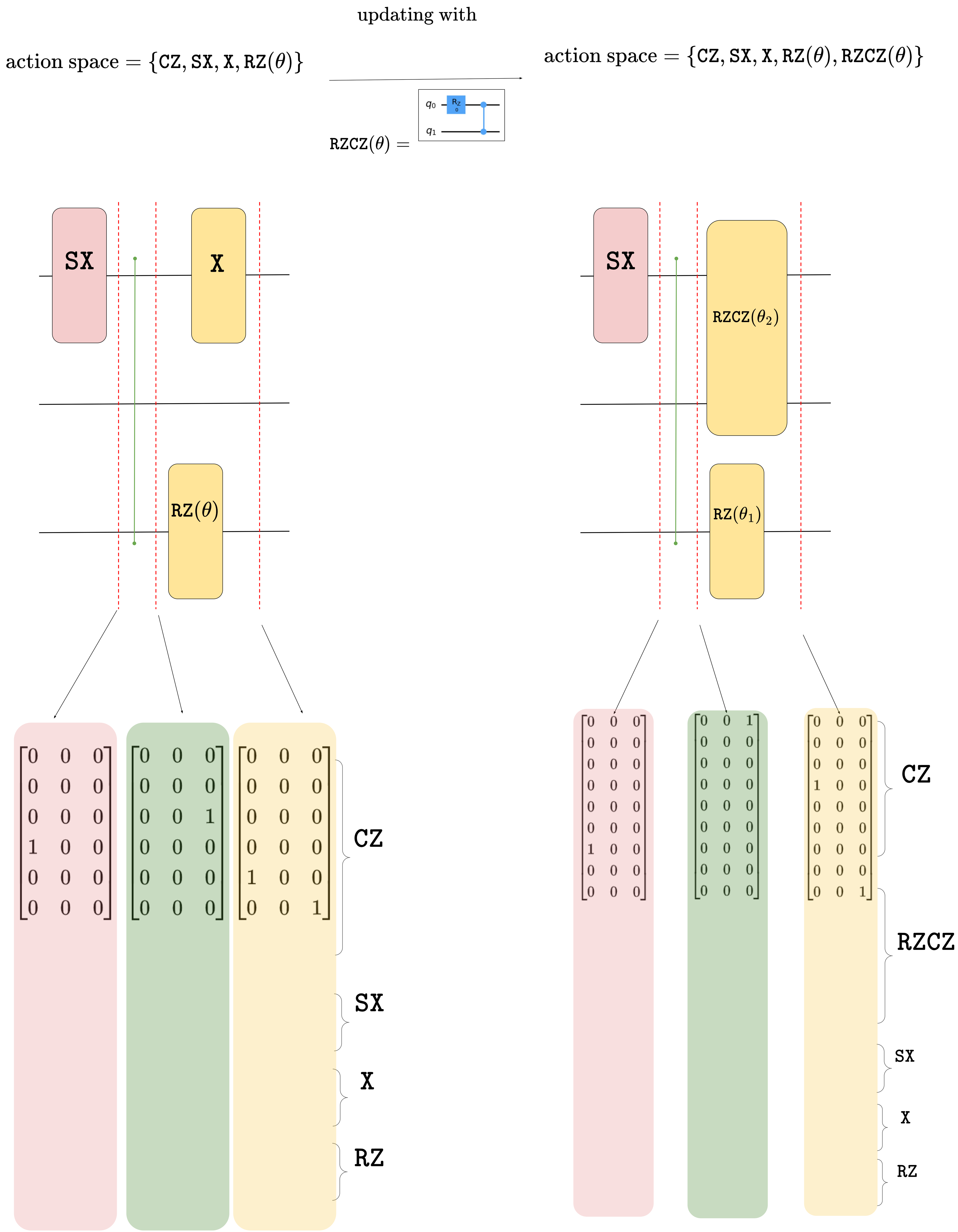
Environment
$|0\rangle$
$U(\vec{\theta})$
$|\psi\rangle$
send
feedback
$\vec{\theta}_{\text{opt}}$
$C(\vec{\theta}) = \langle\psi|H|\psi\rangle$
State preparation
Calculate cost
Action: add gate
X
√X
#1
Policy
State
Reward
Program synthesis:
extend the action space
#1 =
top circuits



Legend:
Elementary gates (RL only)
+ one gate (GRL)
+ two gates (GRL)
2 qubits
3 qubits
a.
b.
c.
d.
e.
f.
g.
h.
Number of successes
$h = 0.05$
$h = 1$
$|E - E_{\mathrm{true}}|$
Performance
Reward threshold
average
range
transverse field
episode


e.
$h = 0.05$
Number of successes
10^3
10^2
10^1
5×10^{-1}
10^{-1}
10^{-3}
10^{-5}
10^{-7}
$|E - E_{\mathrm{true}}|$
f.
$h = 1$
10^2
10^1
10^0
g.
Performance
average
range
$|E - E_{\mathrm{true}}|$
$h = 10^{-3}$
$h = 0.05$
$h = 1$
transverse field
h.
Reward threshold
0
1000
2000
3000
4000
5000
episode


#1
=
X
√X
0
#2
1
=
RZ
0



updating with
action space = {CZ, SX, X, RZ(θ)}
action space = {CZ, SX, X, RZ(θ), RZCZ(θ)}
RZCZ(θ) =
q0
q1
Rz
SX
X
RZ(θ)
SX
RZCZ(θ2)
RZ(θ1)
CZ
SX
X
RZ
CZ
RZCZ
SX
X
RZ