WELCOME TO TUTORIAL

Session 4.1 JanusQ-CT: A Framework for Analyzing Quantum Circuit by Extracting Contextual and Topological Features

ACES Lab
ADVANCED COMPUTING AND EMERGING SERVICE LAB

JanusQ Cloud

浙江大学 计算机科学与技术学院
COLLEGE OF COMPUTER SCIENCE AND TECHNOLOGY
ZHEJIANG UNIVERSITY

https://janusq.github.io/tutorials/

College of Computer Science and Technology,
Zhejiang University

ASPLOS 2024

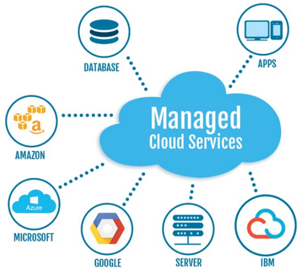
Presenter

Jianwei Yin

zjuyjw@cs.zju.edu.cn

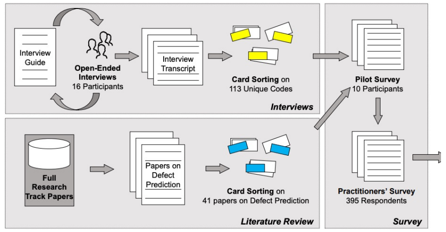
Dr. Jianwei Yin is a full professor at the College of Computer Science, Zhejiang University. He is the vice-dean of the College of Computer Science and Technology. Prof. Yin is served as a PC chair or PC member for international conferences, including ICDCS, ICSOC, ICWS, WISE, etc. He has published more than 100 papers in top international journals and conferences such as MICRO, ASPLOS, ISCA, DAC, TC, TSE, TKDE, TPDS, TSC, TII, TCBY, CHI, ICDE, et al. He led the setup of two international standards/ He won the Best Paper Award in ICSOC 2017 and the SCC 2012 Best Student Paper Award.

Representative works



Edge intelligence: The confluence of edge computing and artificial intelligence

S Deng, J Yin, et al. Citations: 763



Collaborative web service qos prediction with location-based regularization

W Lo, J Yin, et al. Citations: 173

Outline of Presentation

- **Background and challenges**
- QuCT overview
- Upstream model: Circuit feature extraction
- Downstream model 1: Circuit fidelity prediction
- Downstream model 2: Unitary decomposition
- Experiment

Background

ZHEJIANG UNIVERSITY

Compilation of a quantum program

Quantum program → Quantum circuit → Circuit-level compilation → Quantum circuit → Pulse-level compilation → Pulse → Board-level compilation → Board directives → Quantum computer

Circuit-level compilation:

- Input: quantum circuit output: Quantum circuit that satisfies the constraints

Pulse-level compilation:

- Input: quantum circuit output: Pulses received by qubits

Board-level compilation:

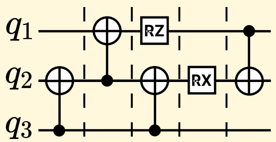
- Input: pulses output: Board directives

Background

Key passes of quantum program compilation

Quantum program $\xrightarrow{\text{Quantum circuit}}$ Circuit-level compilation $\xrightarrow{\text{Quantum circuit}}$ Pulse-level compilation $\xrightarrow{\text{Pulse}}$ Board-level compilation $\xrightarrow{\text{Board directives}}$ Quantum computer

Pass 1: Unitary decomposition



compute

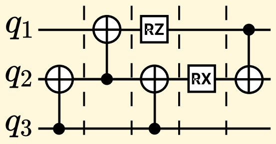
small complexity

$$\begin{bmatrix} 0 & e^{i-\pi/2} & 0 & 0 \\ 0 & 0 & 0 & e^{i-\pi/2} \\ e^{i\pi/2} & 0 & 0 & 0 \\ 0 & 0 & 0 & e^{i\pi/2} \end{bmatrix}$$

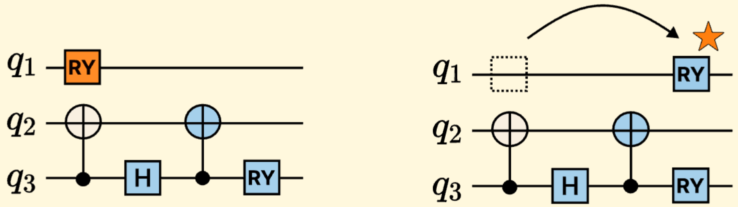
$$\begin{bmatrix} 0 & e^{i-\pi/2} & 0 & 0 \\ 0 & 0 & 0 & e^{i-\pi/2} \\ e^{i\pi/2} & 0 & 0 & 0 \\ 0 & 0 & 0 & e^{i\pi/2} \end{bmatrix}$$

decompose

large complexity



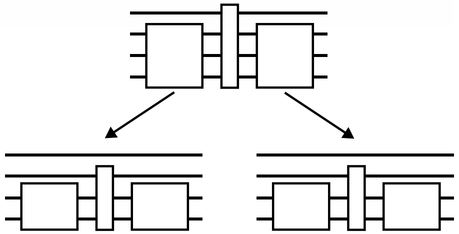
Pass 2: Fidelity prediction and optimization



Optimize the noise while keeping the equivalence of circuits

Challenges of Unitary Decomposition

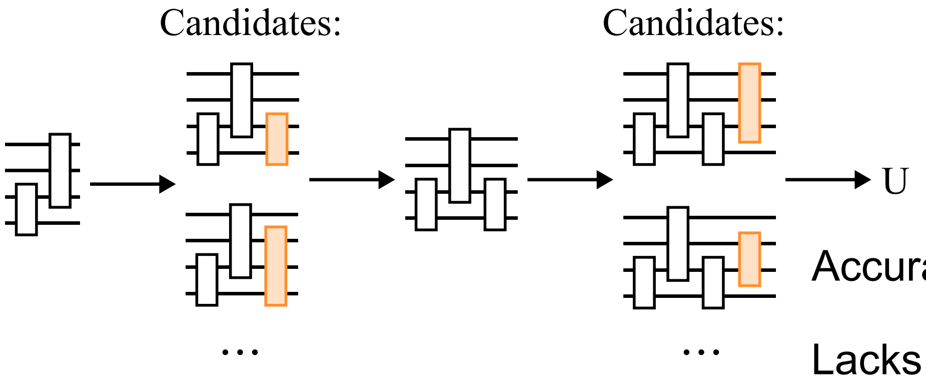
Unitary decomposition



Template-based method

Fast but

Leads to numerous redundant gates



Search-based method

Accurate but

Lacks a heuristic to prune candidate space.

Category	Template-based		Search-based	
Method	CCD [1]	QSD [2]	QFAST [3]	Squander [4]
Time	3.6 s	2.1 s	**511.2 h**	**426.2 h**
#Gate	**3,592**	**3,817**	806	887

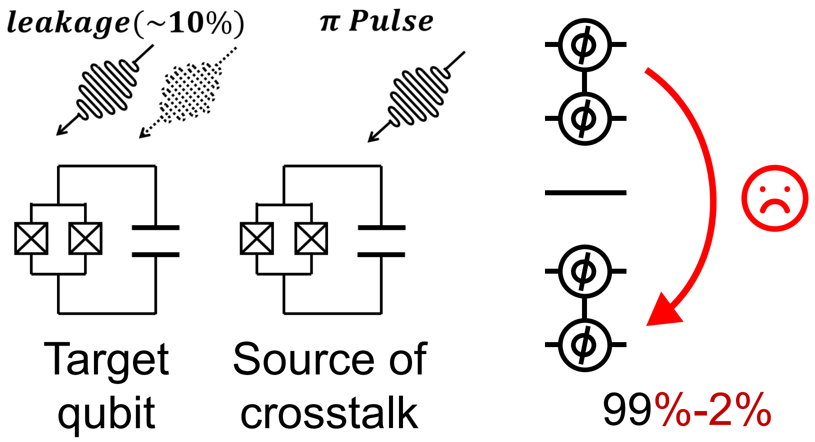
$O(4^N)$ #Gate **$O(4^N)$ Time**

5-qubit unitary decomposition

[1] R. Iten, et al. PRA. 2016
[2] V. Shende, et al. ASP-DAC. 2005
[3] E. Younis, et al. QCE. 2021.
[4] P. Rakyta , et al. Quantum, 2022

Challenges

Fidelity prediction



Related to the circuit structure

Category	RB [5]	XEB [6]	Cycle bench. [7]	Noisy simulat. [8]
Gate-independent error	√	√	√	√
Crosstalk, Pulse distortion	×	×	√	√
Inaccuracy (IBMQ Manila)	**4-28%**	**3-36%**	**2-12%**	**3-17%**

Fidelity prediction

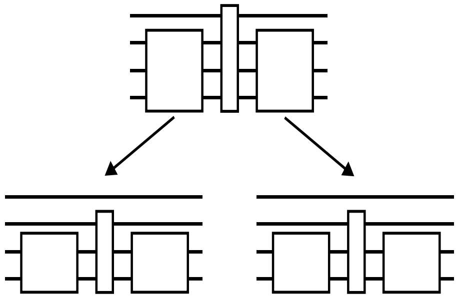
Not one-shot

[5] E. Knill , et al. D. PRA. 2008.
[6] F. Arute, et al. Nature. 2019
[7] A. Erhard, et al. Nature communications. 2019
[8] Isakov, et al ArXiv. 2021.

Current Compilation Methods

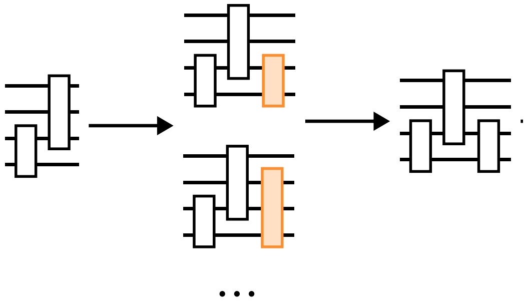
Unitary decomposition

Template-based method



Fast by not accurate:
10-qubit unitary ->
20,000 gate

Search-based method



Accurate but slow:
10-qubit unitary->
one year

Fidelity prediction

Method	Independent noise	Dependent noise	Inaccuracy
RB	√	×	4-28%
XEB	√	×	3-36%
CB	√	√	2-12%
Noisy Simulat.	√	√	3-17%

Fast by inaccurate: cannot model dependent noise

Accurate but slow: require repeated executions

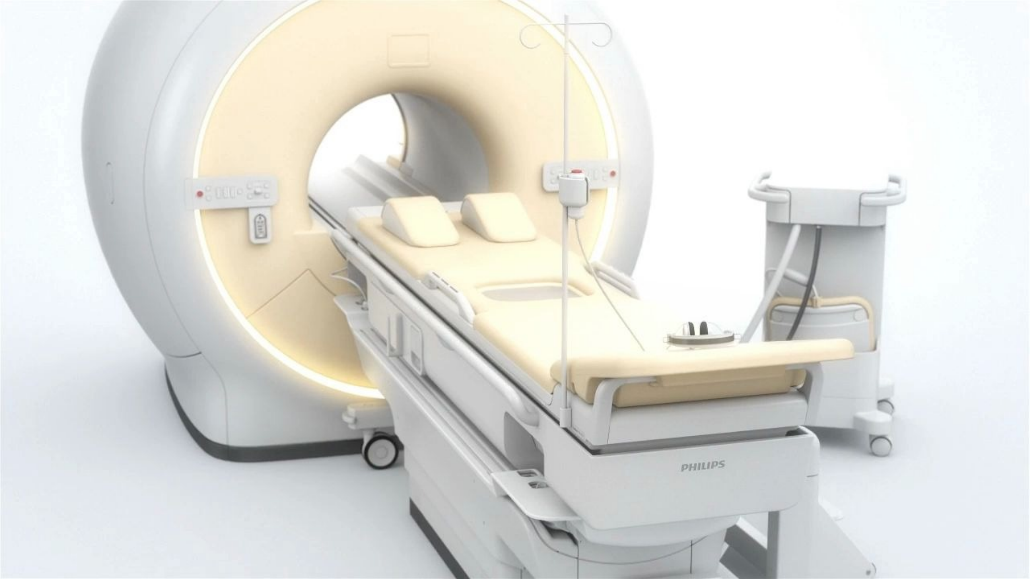
They face a trade-off between the efficiency and accuracy

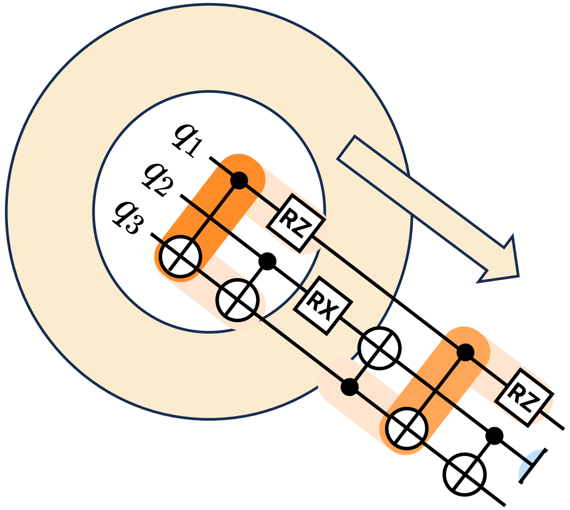
Outline of Presentation

- Background and challenges
- **QuCT overview**
- Upstream model: Circuit feature extraction
- Downstream model 1: Circuit fidelity prediction
- Downstream model 2: Unitary decomposition
- Experiment

QuCT Overview

Origin of the name



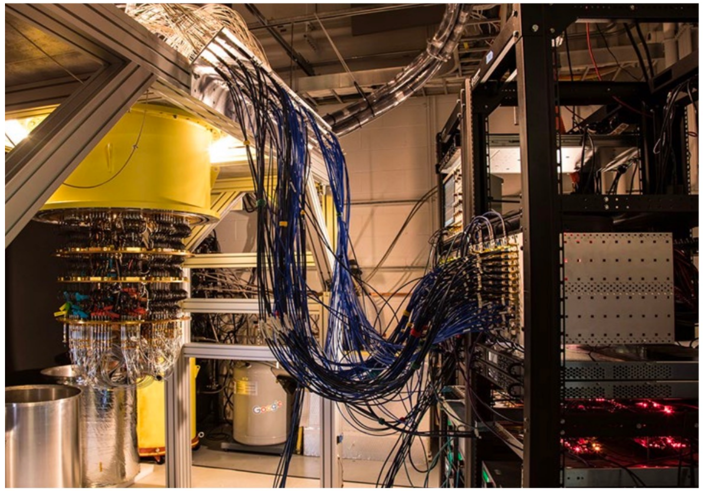


Computerized Tomography

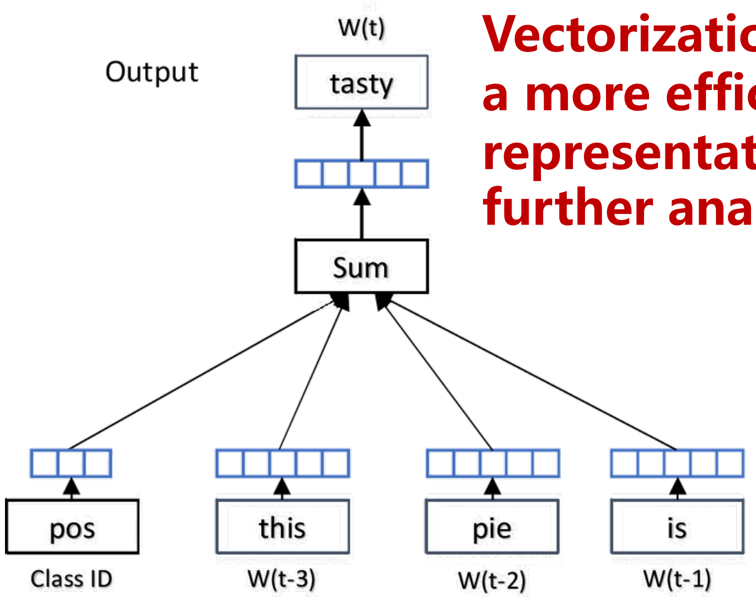
Analyzing Quantum Circuit by Contextual and Topological Features

QuCT Insight

Solution: Implement circuit topology and context-aware gate vectorization



Quantum circuits are implemented via pulses. There are **interactions between wirings of qubits.**



Vectorization is a more efficient representation for further analysis

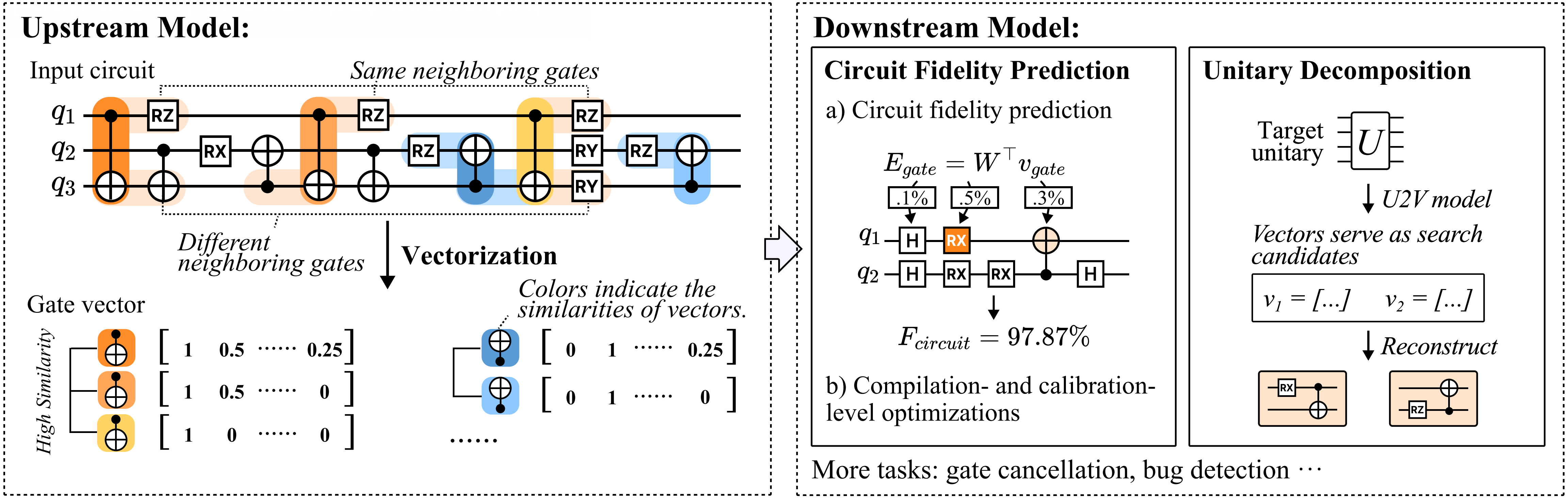
Context extraction is common in **natural language processing (NLP)** and **classical program analysis**

Quantum	NLP
Fidelity prediction	grammar correction
Circuit generation	Test generation

Quantum program analysis and NLP have similar tasks

QuCT Overview

Each model is one-shot generated



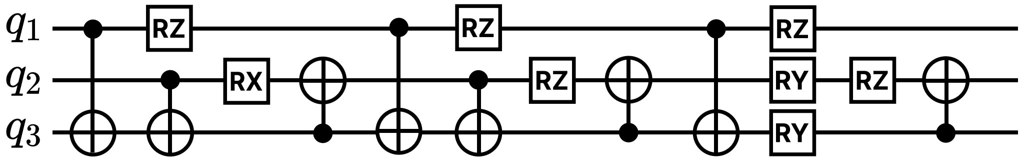
Random walk → Vectorization → Fidelity prediction or Unitary decomposition

QuCT Overview

ZHEJIANG UNIVERSITY

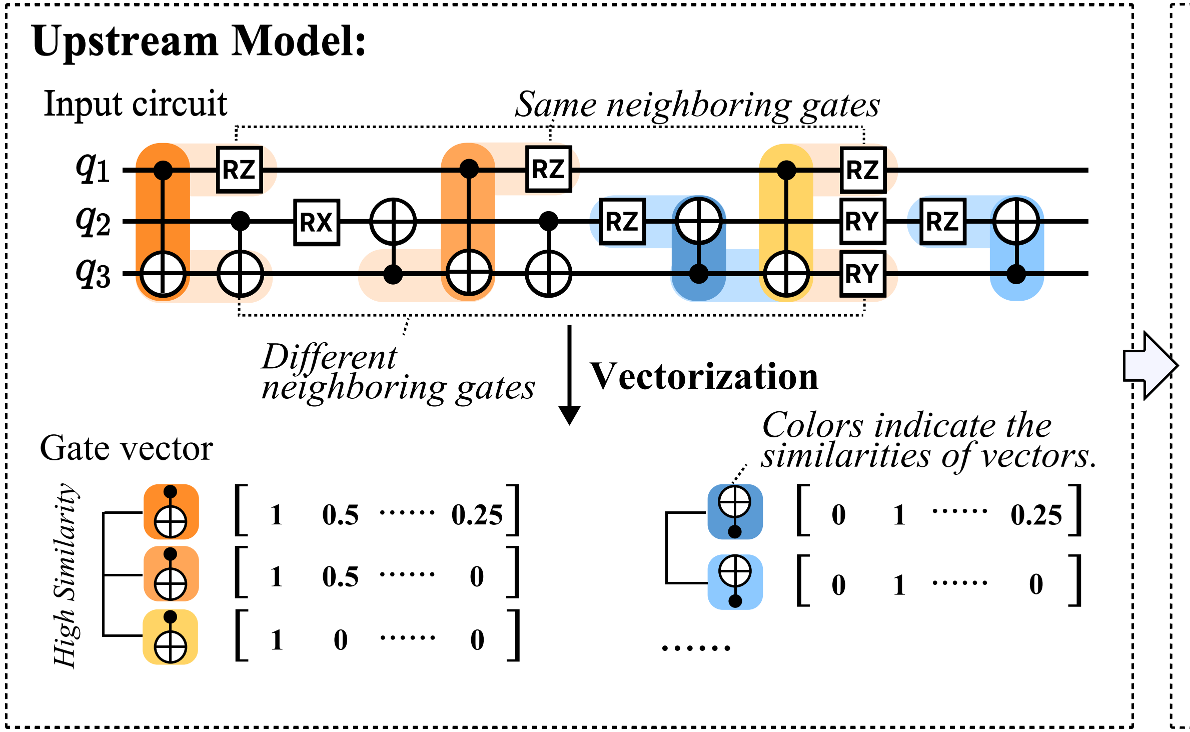
Upstream Model:

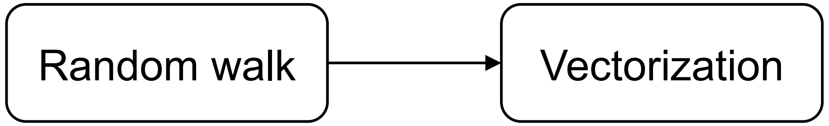
Input circuit



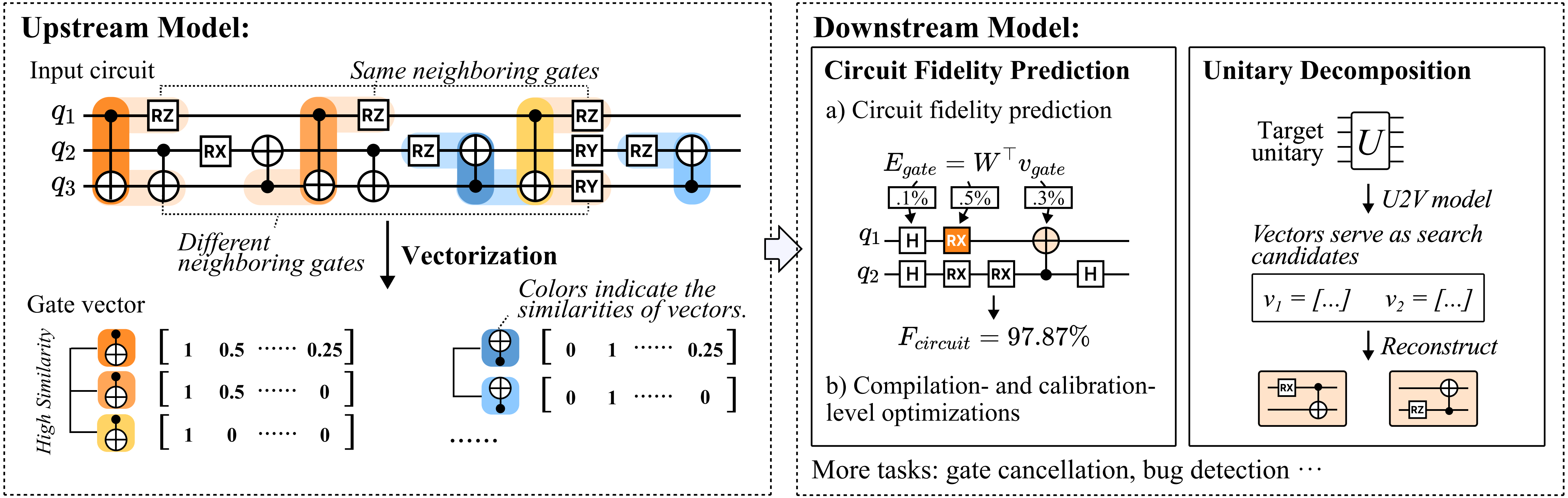
Random walk

QuCT Overview





QuCT Overview



Random walk → Vectorization → Fidelity prediction or Unitary decomposition

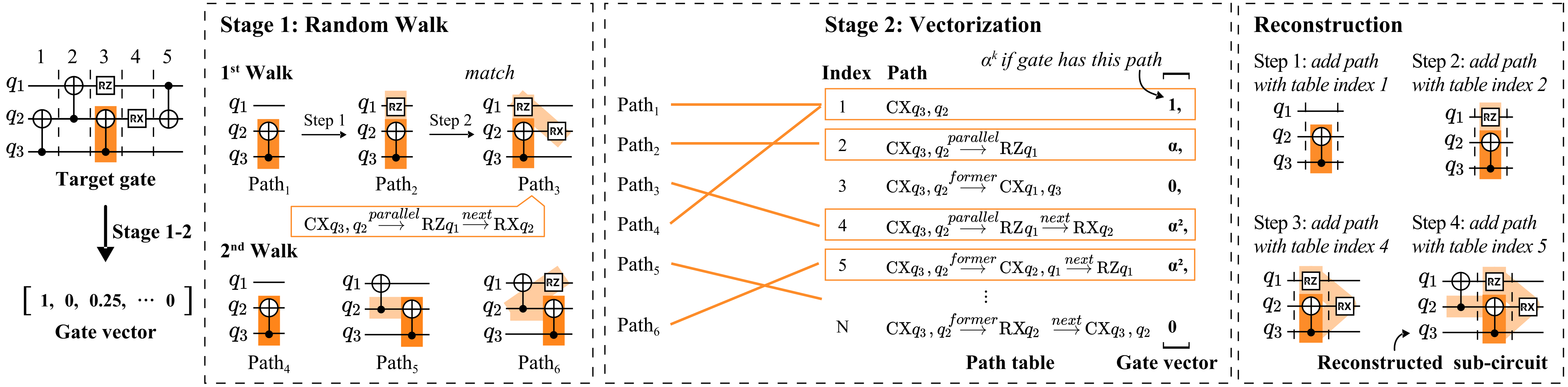
Outline of Presentation

浙江大學
ZHEJIANG UNIVERSITY

- Background and challenges
- QuCT overview
- **Upstream model: Circuit feature extraction**
- Downstream model 1: Circuit fidelity prediction
- Downstream model 2: Unitary decomposition
- Experiment

Upstream Model: Circuit Feature Extraction

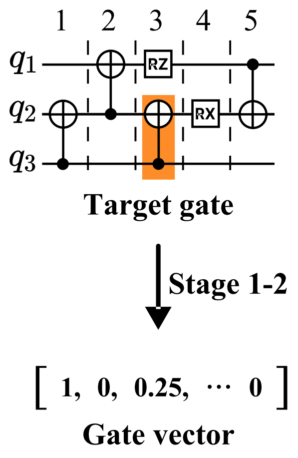
Two-step vectorization flow



Upstream Model: Circuit Feature Extraction

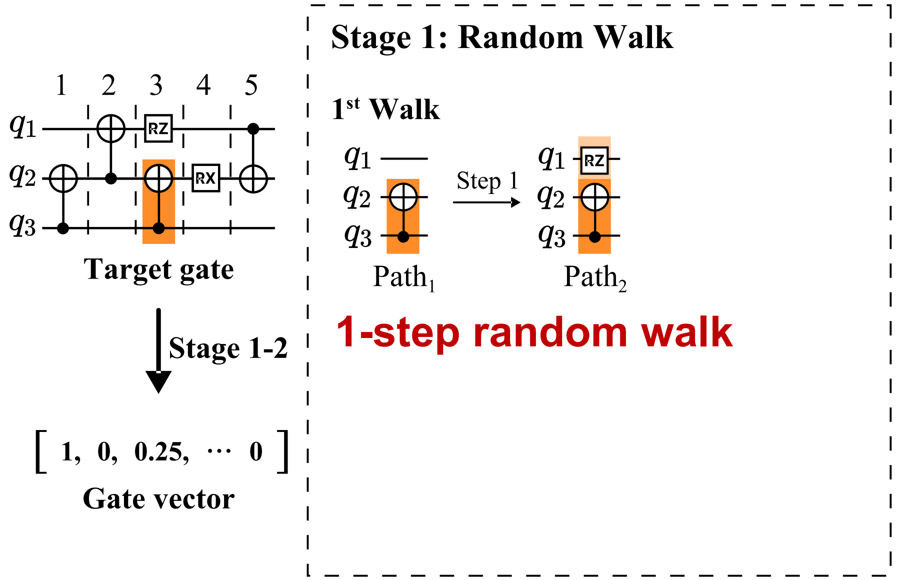
Two-step vectorization flow

For each gate



Upstream Model: Circuit Feature Extraction

Two-step vectorization flow

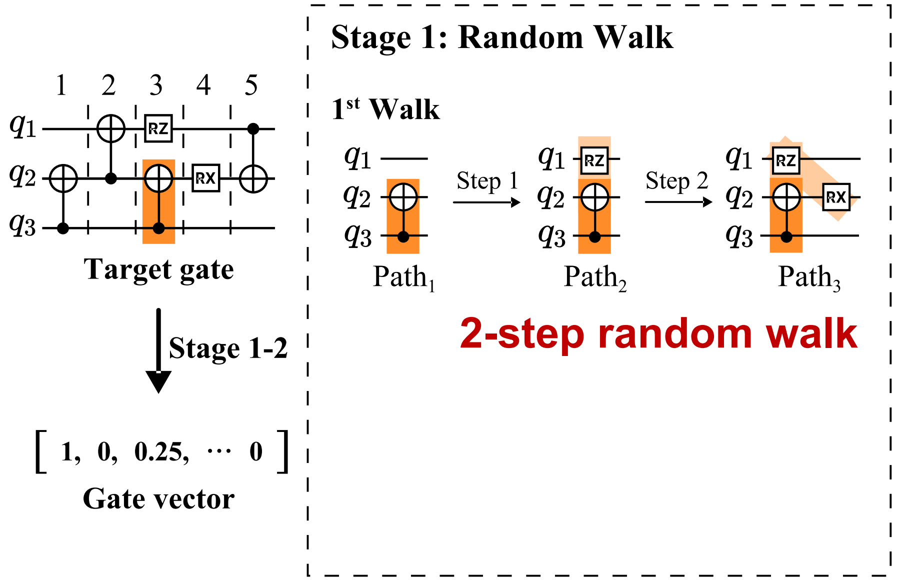


Step 1: Extract features as paths.

Upstream Model: Circuit Feature Extraction

ZHEJIANG UNIVERSITY

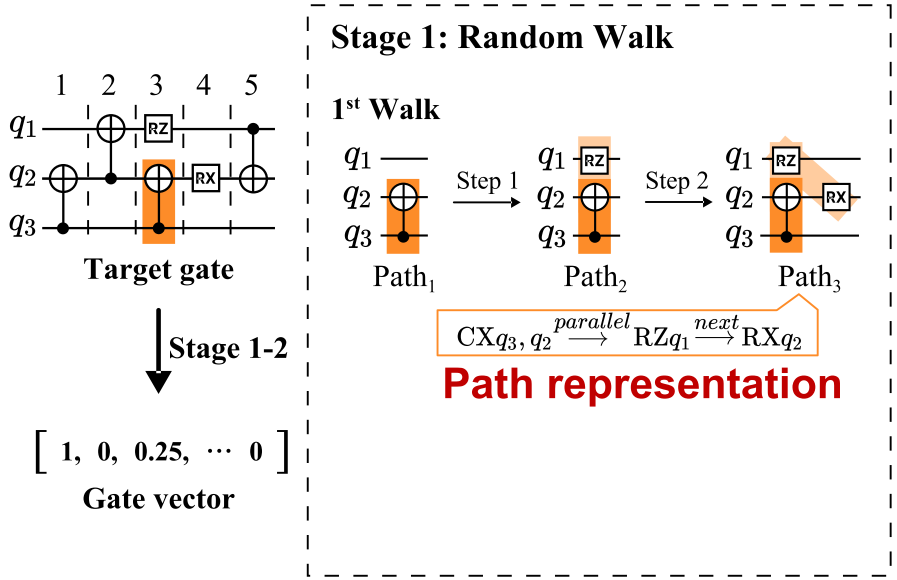
Two-step vectorization flow



Step 1: Extract features as paths.

Upstream Model: Circuit Feature Extraction

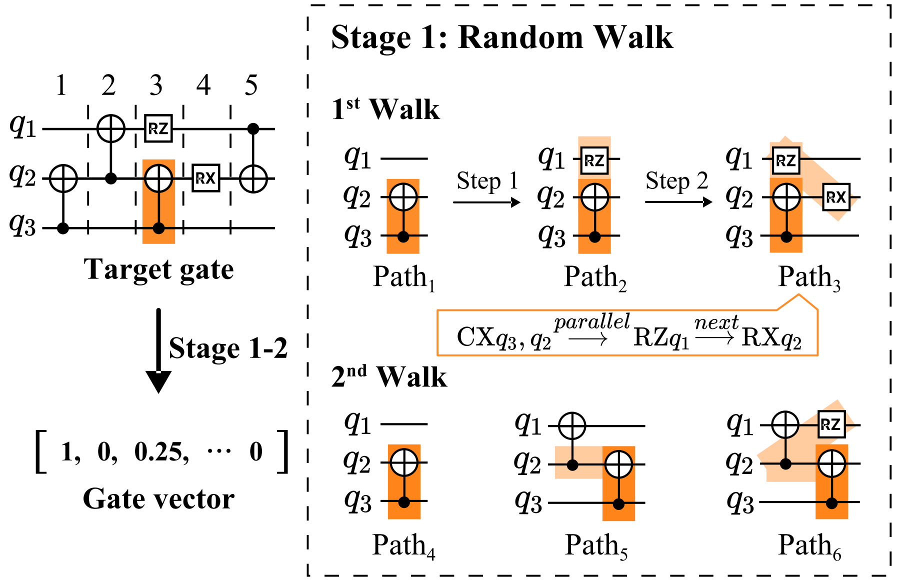
Two-step vectorization flow



Step 1: Extract features as paths.

Upstream Model: Circuit Feature Extraction

Two-step vectorization flow

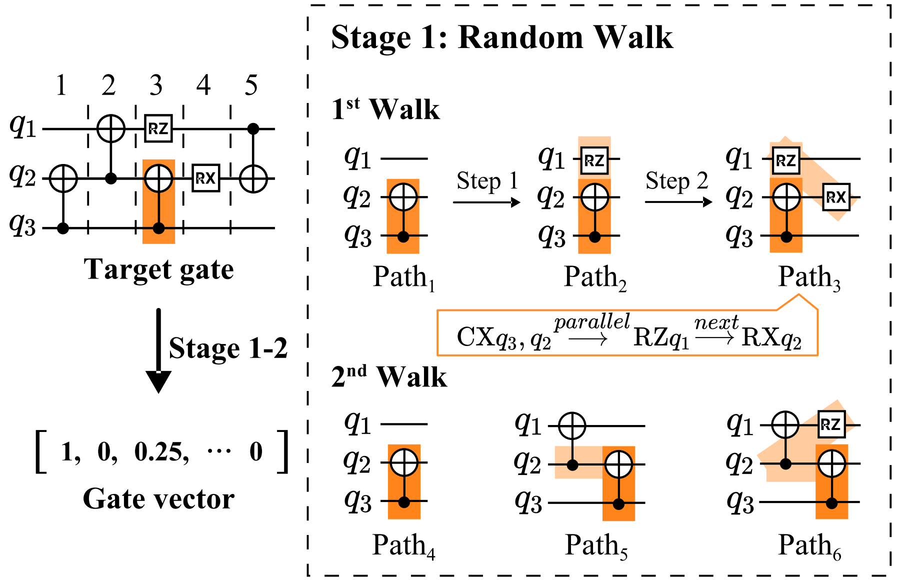


Multiple paths

Step 1: Extract features as paths.

Upstream Model: Circuit Feature Extraction

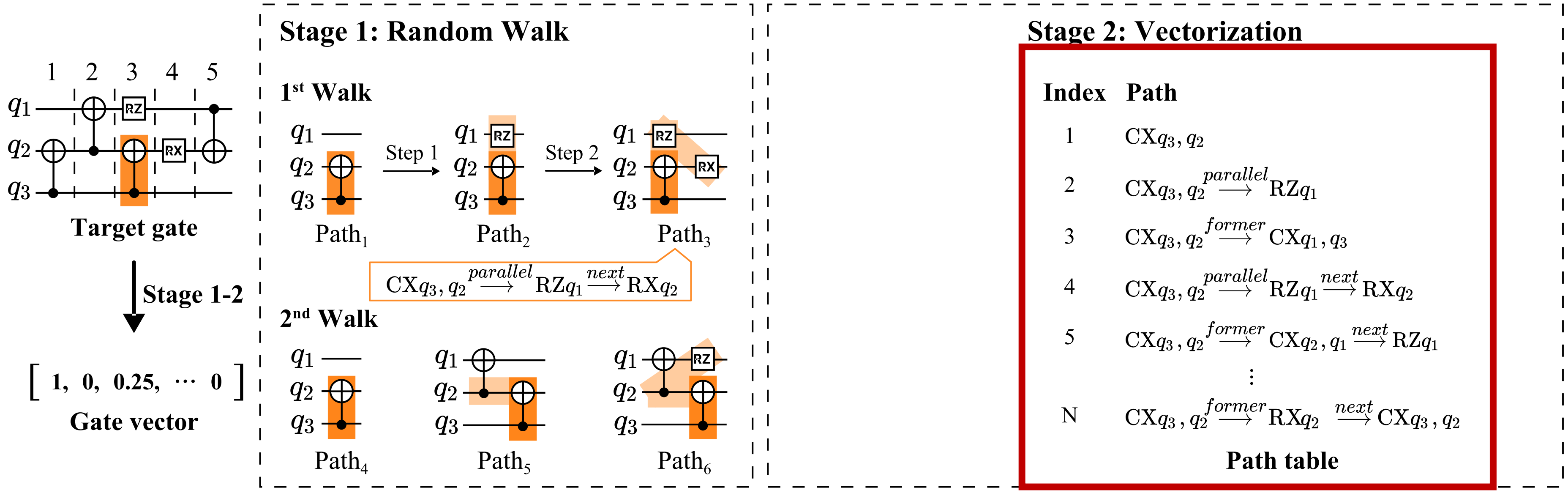
Two-step vectorization flow



Obtain vector.

Upstream Model: Circuit Feature Extraction

Two-step vectorization flow

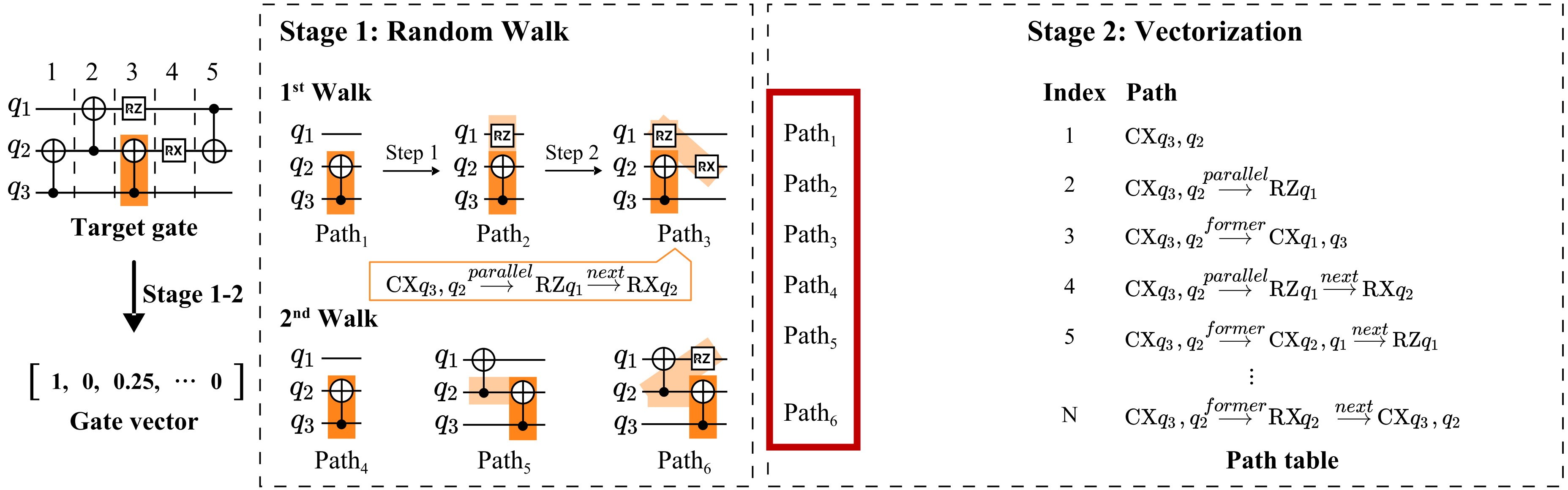


Pre-generated

Obtain vector.

Upstream Model: Circuit Feature Extraction

Two-step vectorization flow



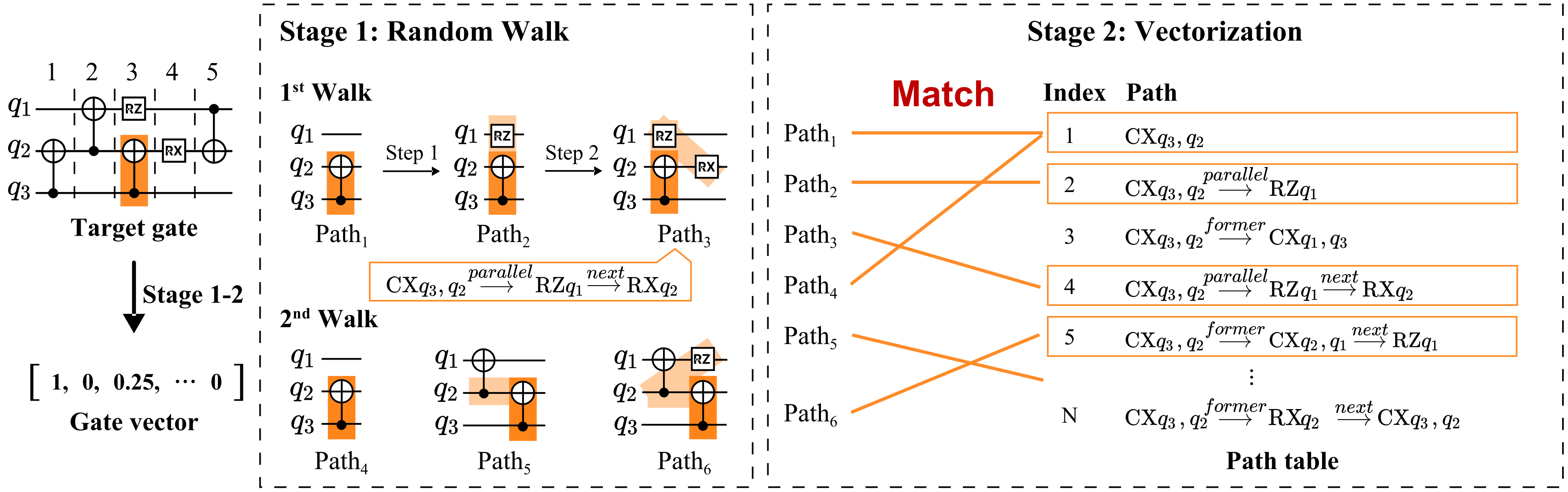
From random walk

Pre-generated

Obtain vector.

Upstream Model: Circuit Feature Extraction

Two-step vectorization flow



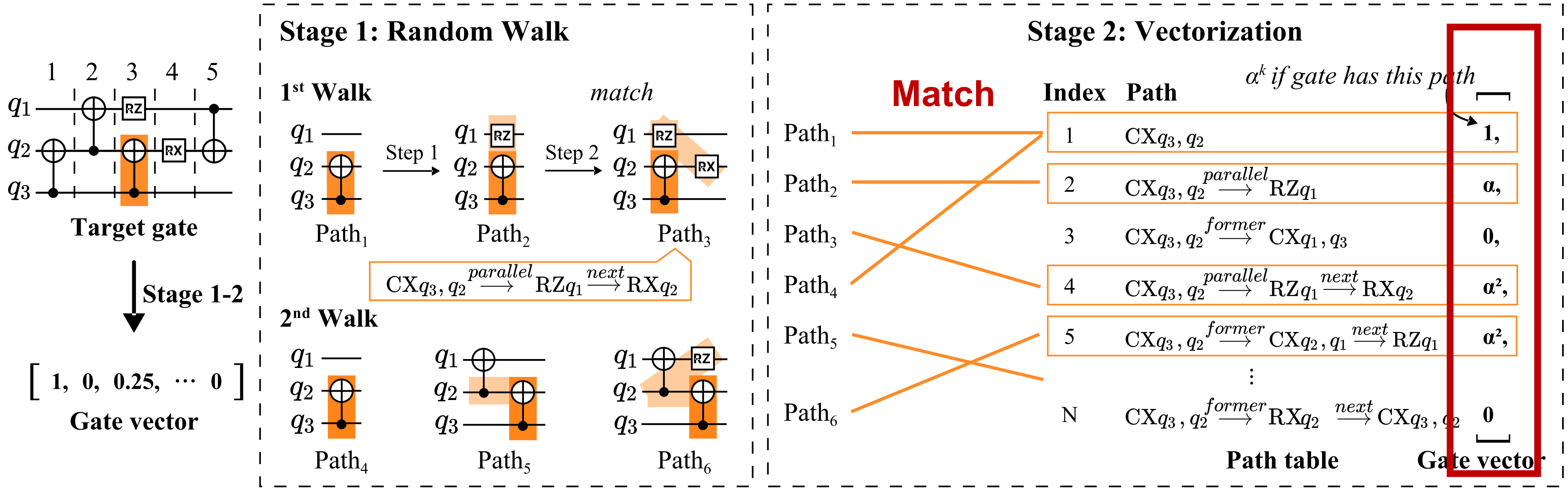
From random walk

Pre-generated

Obtain vector.

Upstream Model: Circuit Feature Extraction

Two-step vectorization flow



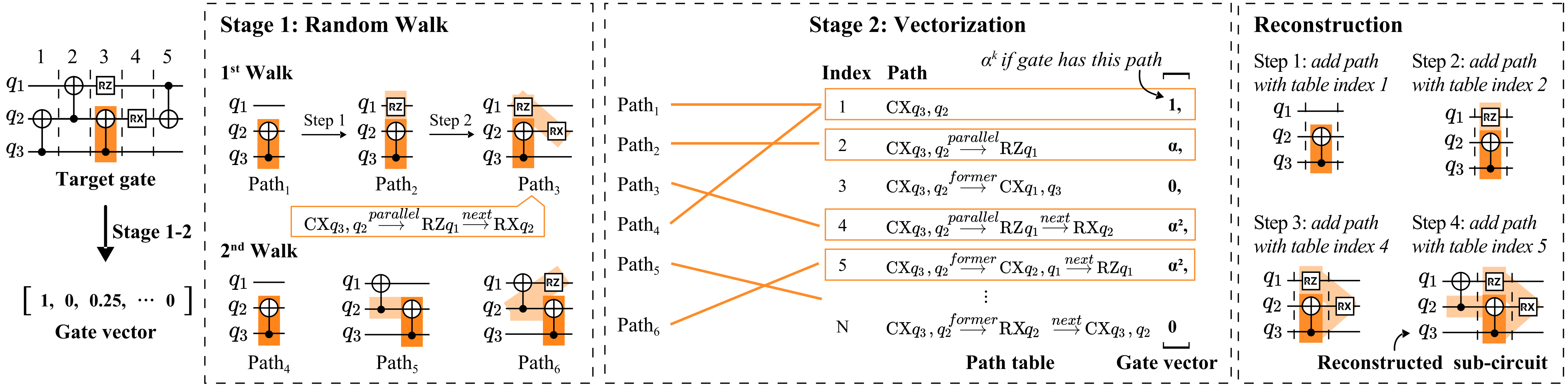
From random walk

Pre-generated

Obtain vector.

Upstream Model: Circuit Feature Extraction

Two-step vectorization flow



Reconstruct circuit.

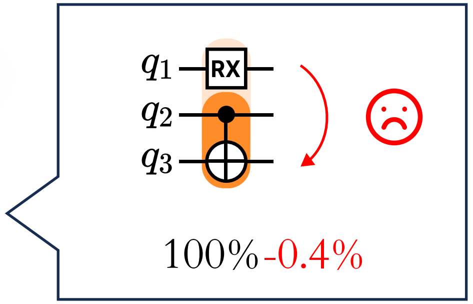
Outline of Presentation

- Background and challenges
- QuCT overview
- Upstream model: Circuit feature extraction
- **Downstream model 1: Circuit fidelity prediction**
- Downstream model 2: Unitary decomposition
- Experiment

Downstream Model 1: Circuit Fidelity Prediction

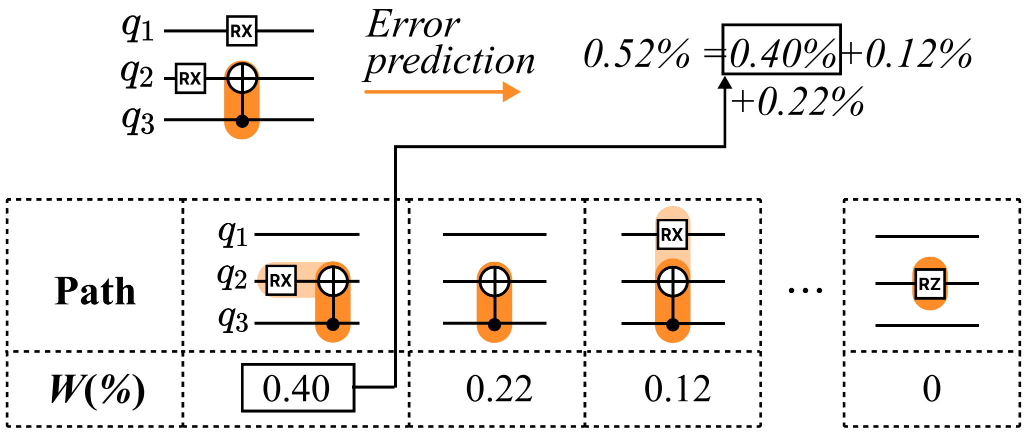
Gate error prediction

$$E(v_i) = W^{\intercal} v_i$$



Model dependent error

v_i : gate vector. W : weight vector obtained via training.



Downstream Model 1: Circuit Fidelity Prediction

Gate error prediction

$$E(v_i) = W^\intercal v_i$$

v_i : gate vector. W : weight vector obtained via training.

Circuit fidelity prediction

$$F_{circuit} = \prod_{g_i \in G} (1 - E(v_i)) \prod_{q \in Q} MF_q$$

G : gate set， Q : qubit set， MF_q : measurement fidelity

The probability that all gates are correct.

Downstream Model 1: Circuit Fidelity Prediction

ZHEJIANG UNIVERSITY

Gate error prediction

$$E(v_i) = W^{\intercal} v_i$$

v_i : gate vector. W : weight vector obtained via training.

Circuit fidelity prediction

$$F_{circuit} = \prod_{g_i \in G} (1 - E(v_i)) \prod_{q \in Q} MF_q$$

G : gate set, Q : qubit set, MF_q : measurement fidelity

Training process of weight vector W :

Obtain fidelity dataset (*circuit,* $F_{ground-truth}$) $\cdots$, $F_{ground-truth}$: ground-truth circuit fidelity on the target quantum device.

$$\min_{W} | F_{circuit} - F_{ground-truth} |$$

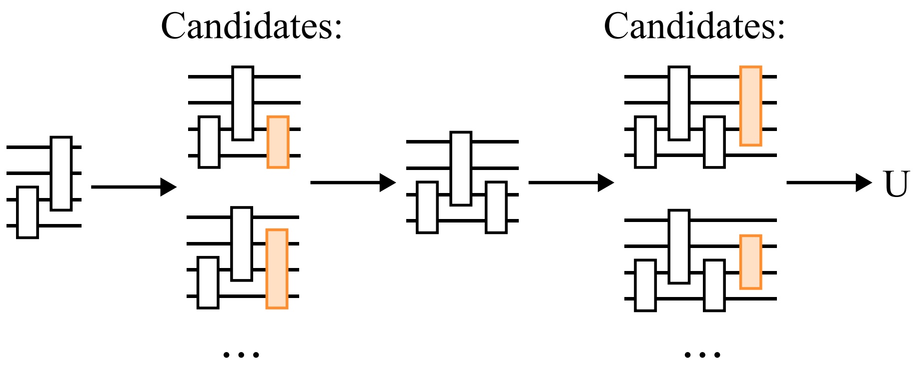
Minimize the distance between the prediction and ground-truth fidelity.

Outline of Presentation

- Background and challenges
- QuCT overview
- Upstream model: Circuit feature extraction
- Downstream model 1: Circuit fidelity prediction
- **Downstream model 2: Unitary decomposition**
- Experiment

Downstream Model 2: Unitary Decomposition

Improve the current search-based method



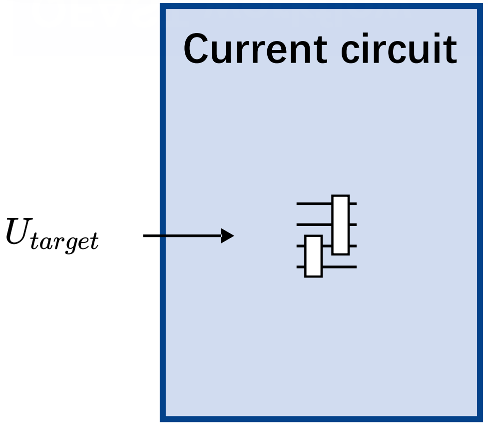
Search-based method

Category	Template-based		Search-based	
Method	CCD [1]	QSD [2]	QFAST [3]	Squander [4]
Time	3.6 s	2.1 s	**511.2 h**	**426.2 h**
#Gate	**3,592**	**3,817**	806	887

5-qubit unitary decomposition

Downstream Model 2: Unitary Decomposition

QFAST workflow

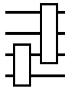


Downstream Model 2: Unitary Decomposition

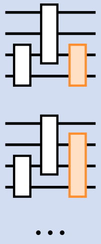
QFAST workflow

Current circuit

U_{target} $\longrightarrow$

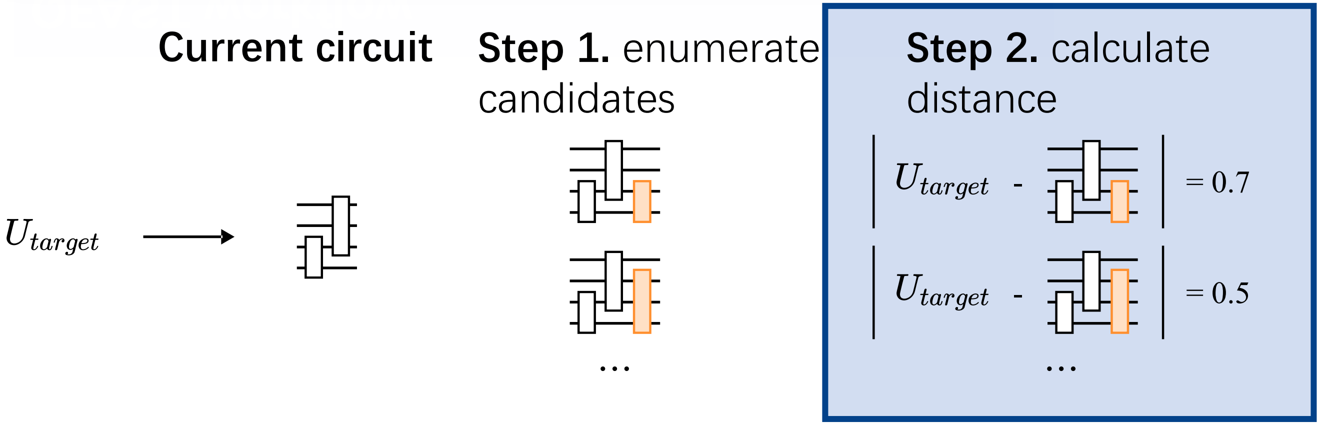


Step 1. enumerate candidates



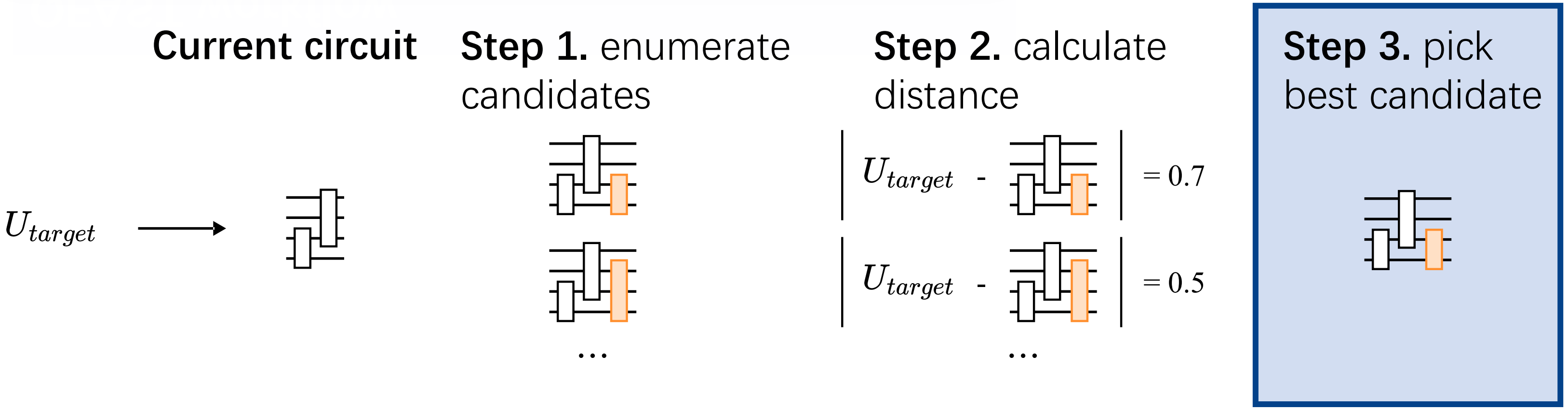
Downstream Model 2: Unitary Decomposition

QFAST workflow



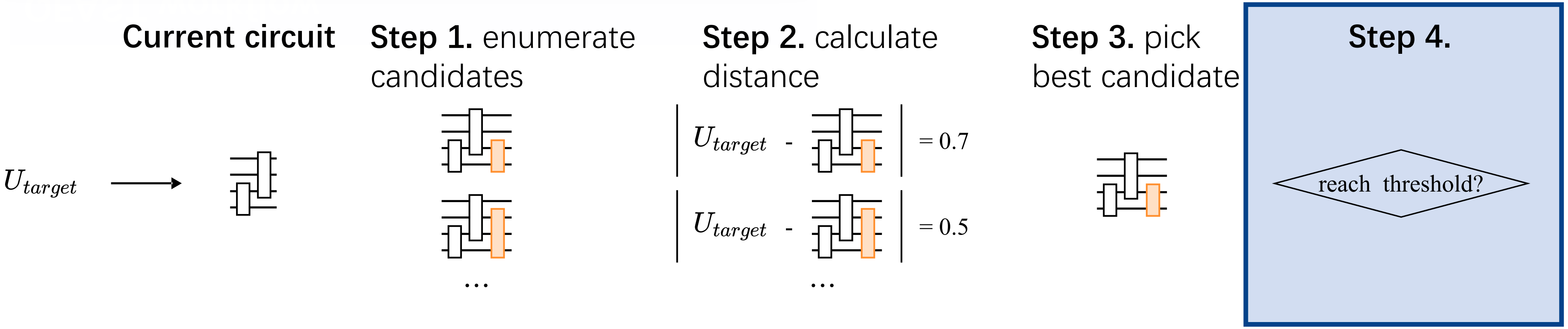
Downstream Model 2: Unitary Decomposition

QFAST workflow



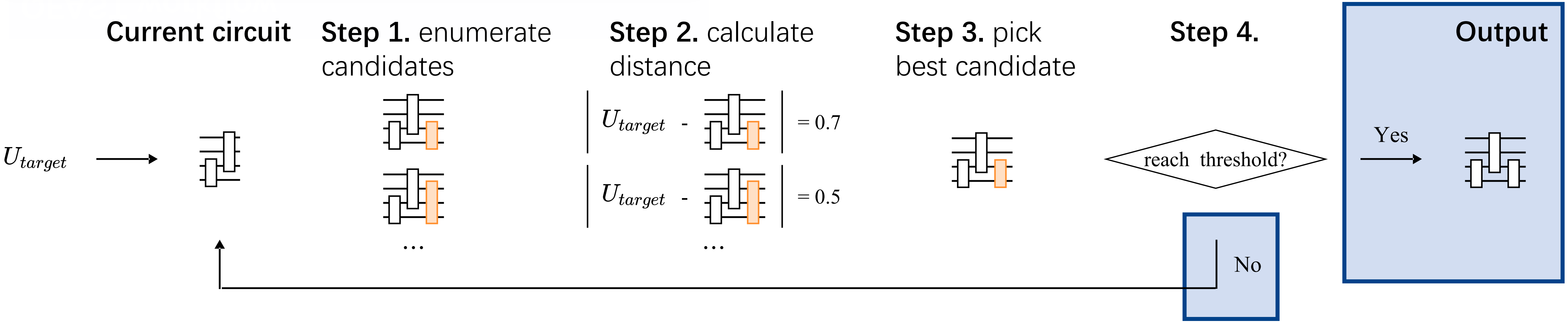
Downstream Model 2: Unitary Decomposition

QFAST workflow



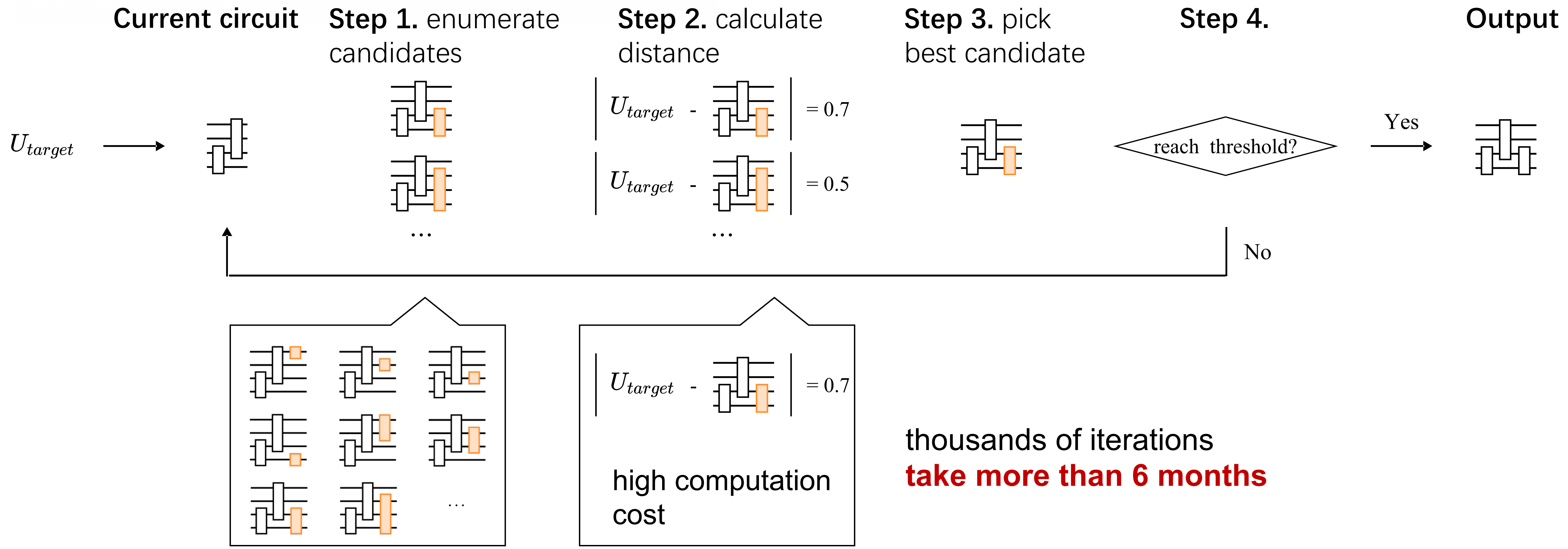
Downstream Model 2: Unitary Decomposition

QFAST workflow



Downstream Model 2: Unitary Decomposition

QFAST workflow



248 candidates for 8 qubits

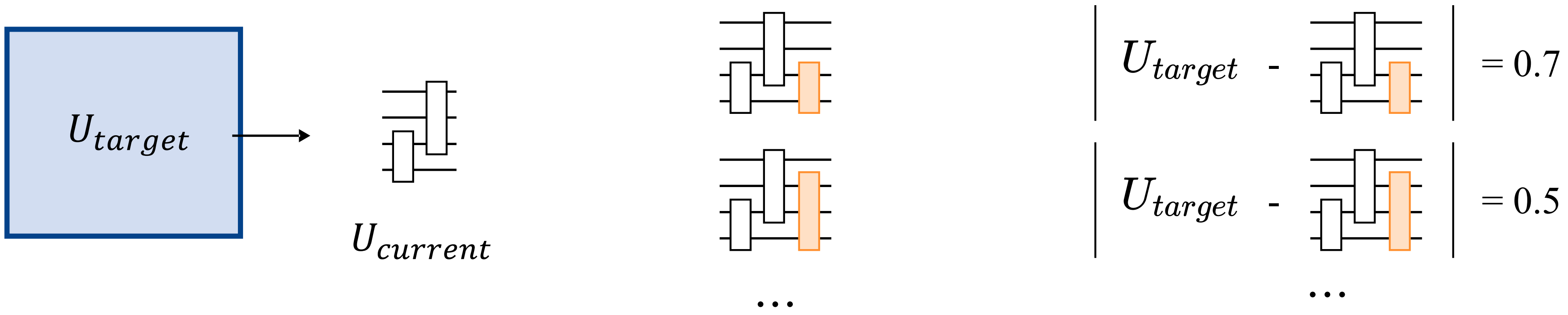
60s/candidate

thousands of iterations
take more than 6 months

Downstream Model 2: Unitary Decomposition

QuCT workflow

Current circuit | **Step 1.** enumerate candidates | **Step 2.** calculate distance | …



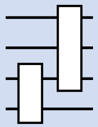
Candidates and should equal $U_{target}U_{current}^{-1}$

Downstream Model 2: Unitary Decomposition

QuCT workflow

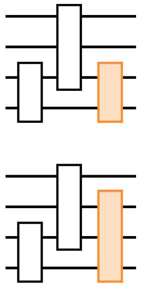
Current circuit

U_{target} →



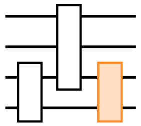
$U_{current}$

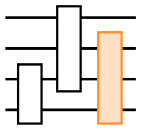
Step 1. enumerate candidates



…

Step 2. calculate distance

$\left| U_{target} - \right.$  $\left. \right| = 0.7$

$\left| U_{target} - \right.$  $\left. \right| = 0.5$

…

…

Candidates and should equal $U_{target}U_{current}^{-1}$

Downstream Model 2: Unitary Decomposition

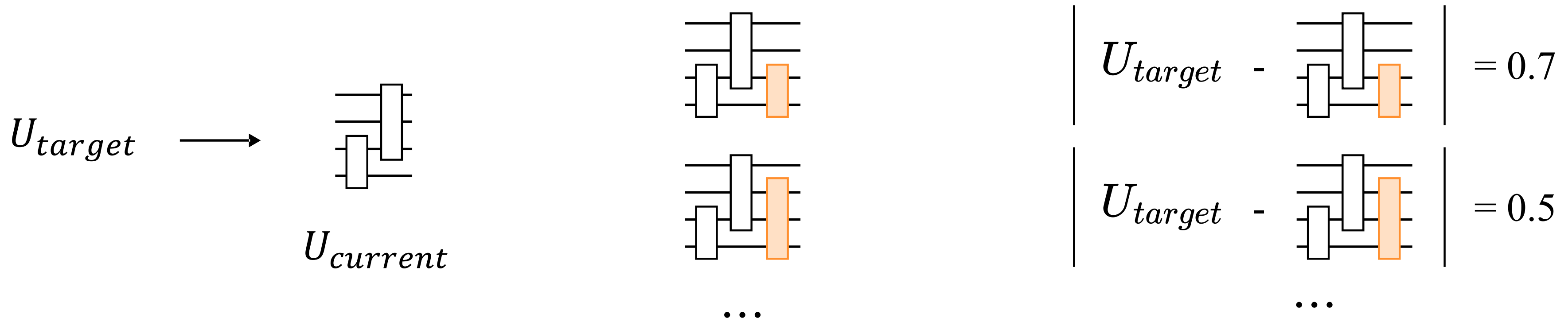
QuCT workflow

Current circuit

Step 1. enumerate candidates

Step 2. calculate distance

…

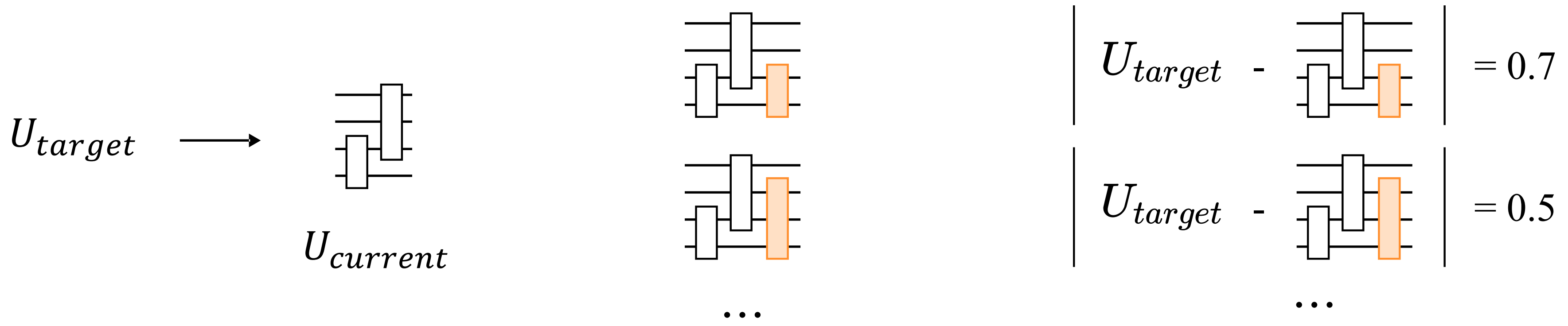


Candidates and should equal $U_{target}U_{current}^{-1}$

Downstream Model 2: Unitary Decomposition

QuCT workflow

Current circuit **Step 1.** enumerate candidates **Step 2.** calculate distance …



Candidates and should equal $U_{target}U_{current}^{-1}$

Instead of exhaustive search, QuCT only try the candidates that have high probability of equaling $U_{target}U_{current}^{-1}$

Downstream Model 2: Unitary Decomposition

QuCT workflow

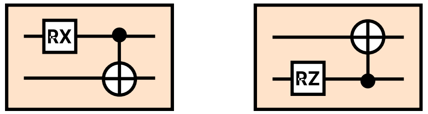
$$U_{target}U_{current}^{-1}$$

↓ U2V model

Gate vectors serve as search candidates

$v_1 = [...]$ $v_2 = [...]$

↓ Reconstruct

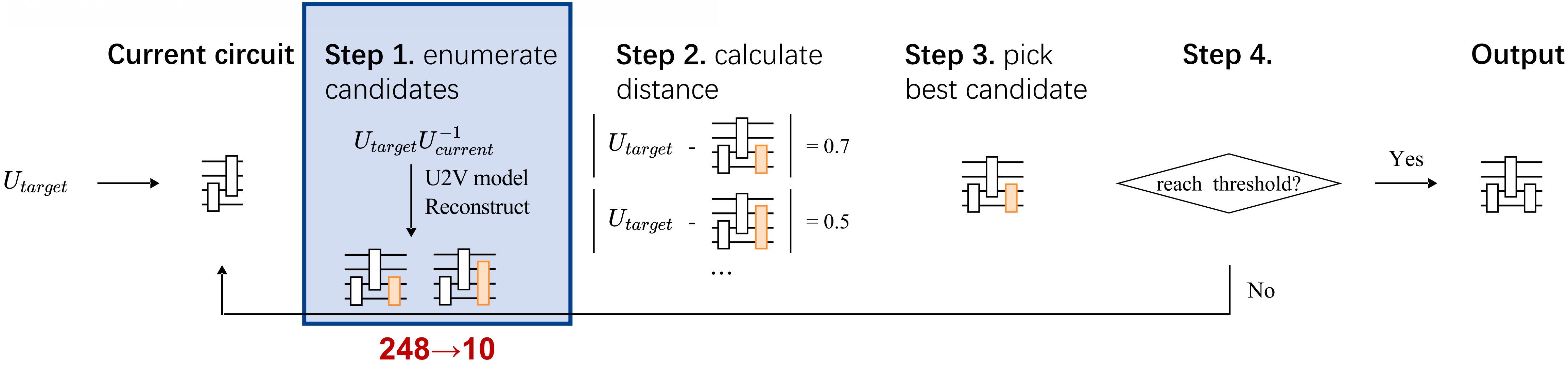


U2V model: a random forest model, trained by the pre-generated decomposition results.

Use gate vectors to construct good candidates.

Downstream Model 2: Unitary Decomposition

QuCT workflow



Comparison to the template-based method: template-based approach has smaller design space, as it can only selects candidates from a limited-size template library.

1.8× speedup and 1.6 × gate reduction compared to the template-based method.

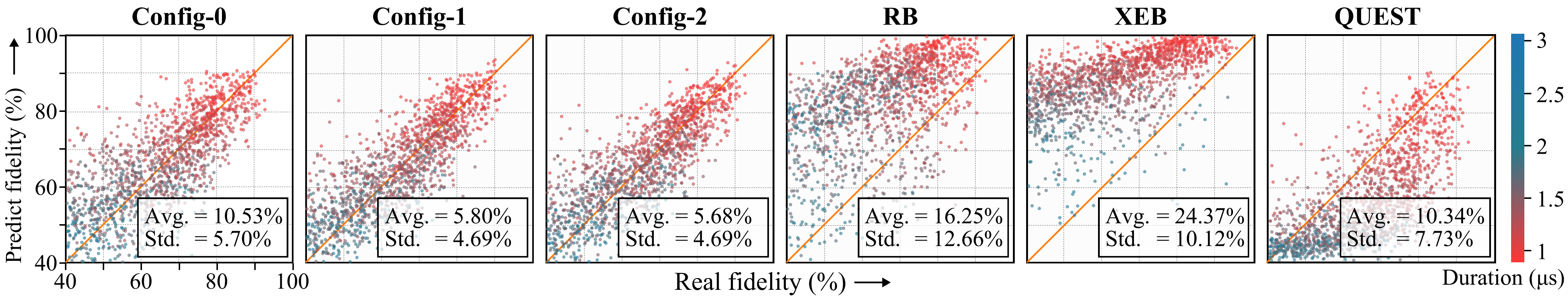
Outline of Presentation

Experiment of Fidelity Prediction

Config.	Upstream model			Downstream model			
	Qubit	N_{step}	Path table	W	Fidelity dataset	Training time	Test dataset
Config-0	18	0	45	45	2,000	1.2 min	2,000
Config-1	18	1	3,420	3,420	2,000	10.5 min	2,000
Config-2	18	2	11.5k	11.5k	2,000	95.6 min	2,000

RB: E. Knill , et al. D. PRA. 2008.
XEB: F. Arute, et al. Nature. 2019
QUEST: H. Wang, et al ICCAD. 2022.

QuCT configuration

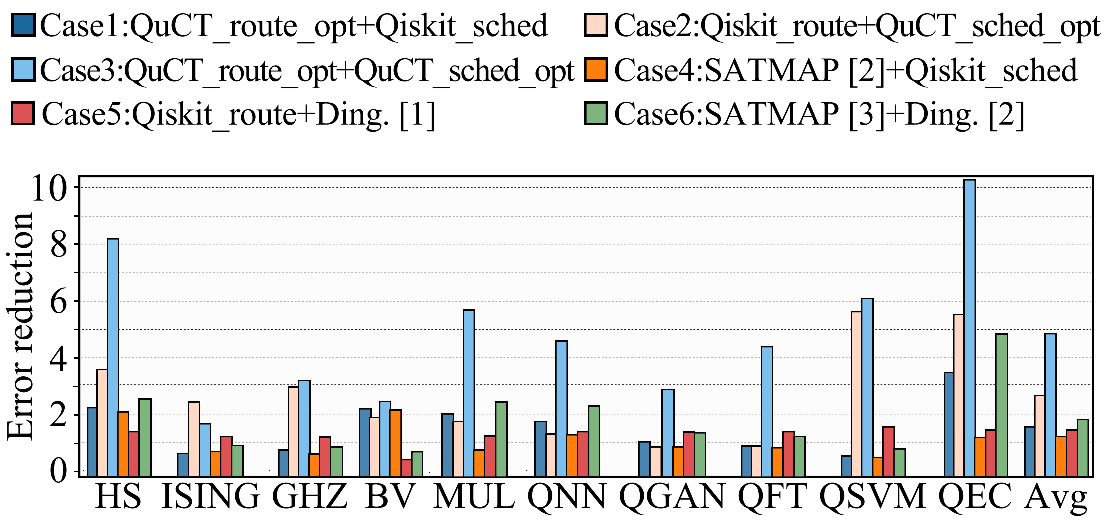


4.2× inaccuracy reduction on a 18-qubit custom superconducting quantum computer.

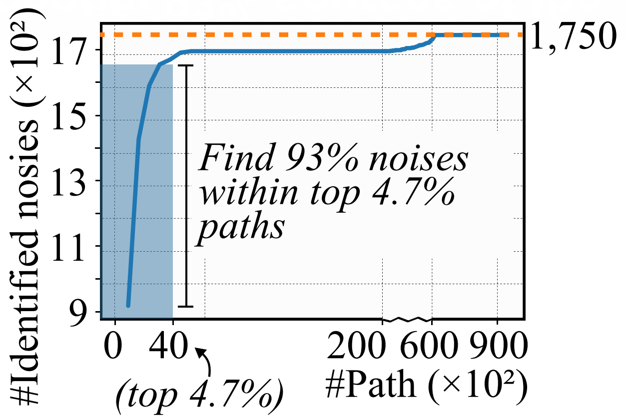
Experiment of Fidelity Optimization

浙江大學 ZHEJIANG UNIVERSITY

Up to 9.2× error reduction



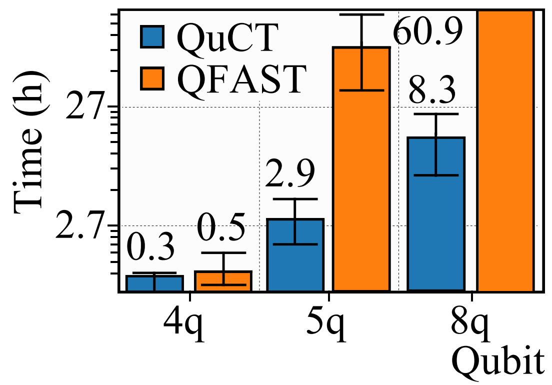
Identify 93% sources of the noise on the simulator.



[1] Y. Ding, et al. MICRO. 2020
[2] A. Molavi, et al. MICRO. 2022

Experiment of Unitary Decomposition

		Metric	CCD [1]	QSD [2]	QFAST [3]	QuCT	Improve
Random	4q	#Gate	1,080.5	859.4	226.3	206.3	1.1X
		Depth	170.8	120.1	74.1	67.4	1.1X
		Time	1.5 s	0.5 s	8.2 h	1.8h	4.6X
	5q	#Gate	3,592.9	3,817.2	887.5	688.1	1.3X
		Depth	465.5	528.1	294.1	227.7	1.3X
		Time	3.6 s	2.1 s	426.2 h	9.2 h	46.3X
	8q	#Gate	7.8×10^4	9.3×10^4	-	2.1×10^4	-
		Depth	3.1×10^4	3.6×10^4	-	1.3×10^4	-
		Time	111.5 s	33.0 s	> 1 year	144.4h	>59.8X
Benchmark	4q	#Gate	236.9	270.5	41.0	40.6	1.0X
		Depth	169.5	193.5	17.0	18.2	0.9X
		Time	0.9 s	0.7 s	1,341.8 s	619.9 s	2.2X
	5q	#Gate	405.0	472.0	181.3	165.2	1.1X
		Depth	302.3	348.7	93.6	91.0	1.0X
		Time	0.8 s	0.5 s	23.3 h	2043.0 s	37.5X
	8q	#Gate	8.1×10^4	8.9×10^4	-	3,392.0	-
		Depth	6.4×10^4	7.6×10^4	-	1,948.0	-
		Time	158.2 s	262.5 s	> 6 months	26.1 h	>165.3X



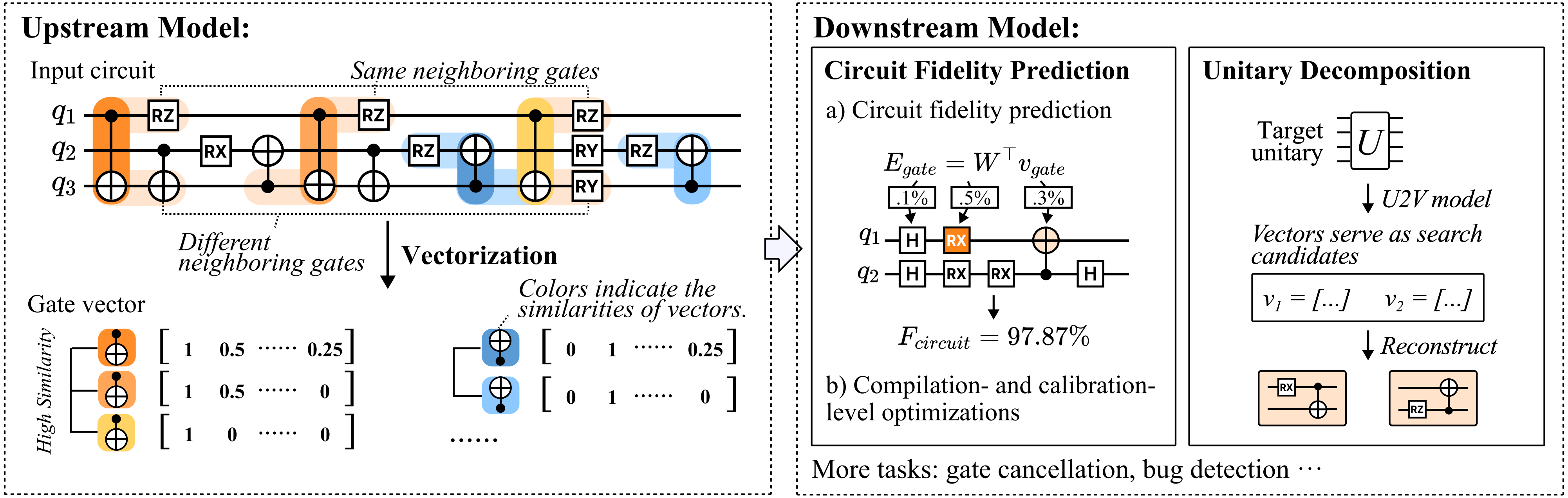
① 46.3× speedup on average for 5-qubit unitaries；

② 8.3 hours for 8-qubit unitaries with multi-threading.

[1] CCD R. Iten, et al. PRA. 2016
[2] QSD V. Shende, et al. ASP-DAC. 2005
[3] QFAST E. Younis,, et al. QCE. 2021.

Conclusion

- Random walk-based method to extract contextual and topological circuit feature.
- Accurate circuit fidelity prediction via modeling gate interactions.
- Fast unitary decomposition via pruning candidate space.



Thanks for listening

QuCT: A Framework for Analyzing Quantum Circuit by Extracting Contextual and Topological Features

Siwei Tan; Congliang Lang; Liang Xiang; Shudi Wang; Xinghui Jia; Ziqi Tan; Tingting Li; Jieming Yin; Yongheng Shang, Andre Python, Liqiang Lu*, Jianwei Yin*