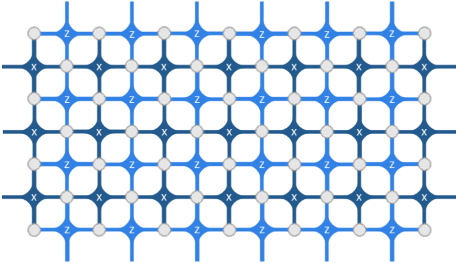
NISQ+: Boosting quantum computing power by approximating quantum error correction

Yichao Yu

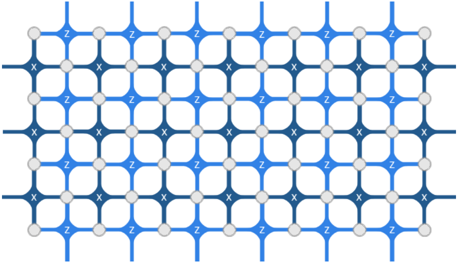
Ni Group

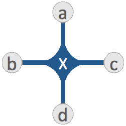
Apr. 26, 2020

Stabilizer operators

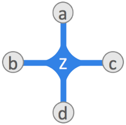


Stabilizer operators



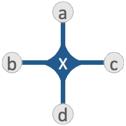


$$X = \prod_{i=a,b,c,d} \sigma_i^x$$



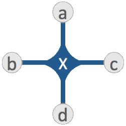
$$Z = \prod_{i=a,b,c,d} \sigma_i^z$$

Error and stabilizer



$$X = \prod_{i=a,b,c,d} \sigma_i^x$$

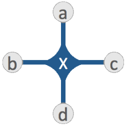
Error and stabilizer



$$X = \prod_{i=a,b,c,d} \sigma_i^x$$

Qubit state: $X|\psi\rangle = |\psi\rangle$
Error: σ_a^z

Error and stabilizer

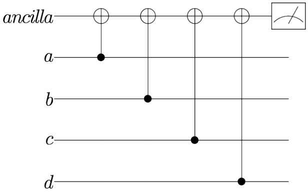


$$X = \prod_{i=a,b,c,d} \sigma_i^x$$

Qubit state: $X|\psi\rangle = |\psi\rangle$
Error: σ_a^z

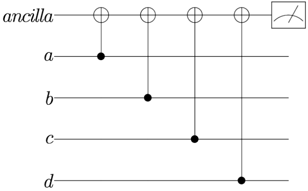
$$X\sigma_a^z|\psi\rangle = -\sigma_a^z X|\psi\rangle = -\sigma_a^z|\psi\rangle$$

Gate implementation of stabilizer: Z



$$Z = \prod_{i=a,b,c,d} \sigma_i^z$$

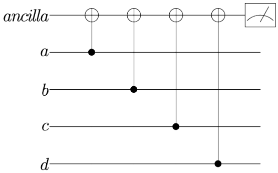
Gate implementation of stabilizer: Z



$$Z = \prod_{i=a,b,c,d} \sigma_i^z$$

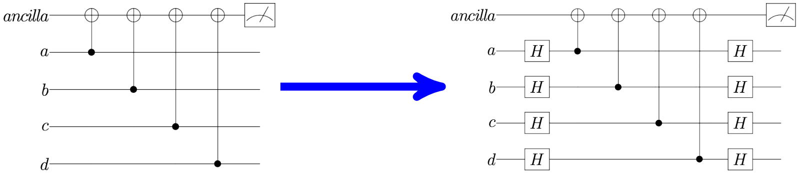
a	b	c	d	ancilla	$\langle Z \rangle$
$\|0\rangle$	$\|0\rangle$	$\|0\rangle$	$\|0\rangle$	$\|0\rangle$	1
$\|1\rangle$	$\|0\rangle$	$\|0\rangle$	$\|0\rangle$	$\|1\rangle$	-1
$\|1\rangle$	$\|1\rangle$	$\|0\rangle$	$\|0\rangle$	$\|0\rangle$	1
$\|1\rangle$	$\|1\rangle$	$\|1\rangle$	$\|0\rangle$	$\|1\rangle$	-1
$\|1\rangle$	$\|1\rangle$	$\|1\rangle$	$\|1\rangle$	$\|0\rangle$	1

Gate implementation of stabilizer: X



$$Z = \prod_{i=a,b,c,d} \sigma_i^z$$

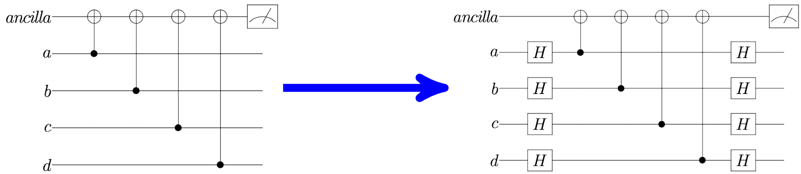
Gate implementation of stabilizer: X



$$Z = \prod_{i=a,b,c,d} \sigma_i^z$$

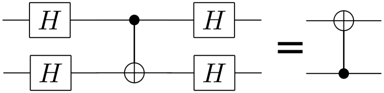
$$X = \prod_{i=a,b,c,d} \sigma_i^x$$

Gate implementation of stabilizer: X

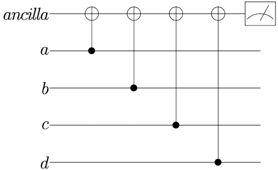


$$Z = \prod_{i=a,b,c,d} \sigma_i^z$$

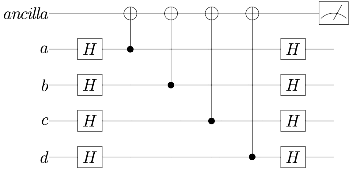
$$X = \prod_{i=a,b,c,d} \sigma_i^x$$



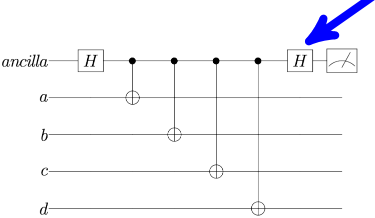
Gate implementation of stabilizer: X

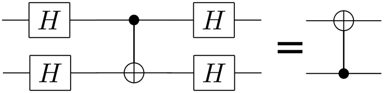


$$Z = \prod_{i=a,b,c,d} \sigma_i^z$$

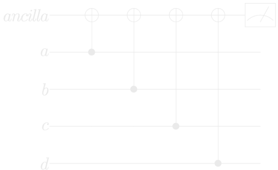


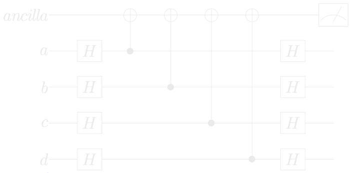
$$X = \prod_{i=a,b,c,d} \sigma_i^x$$





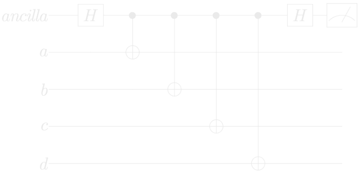
Gate implementation of stabilizer: X

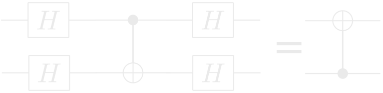




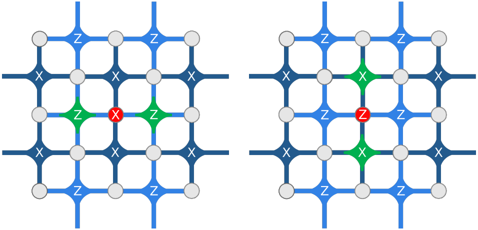
$$Z = \prod_{i=a,b,c,d} \sigma_i^z$$

$$X = \prod_{i=a,b,c,d} \sigma_i^x$$

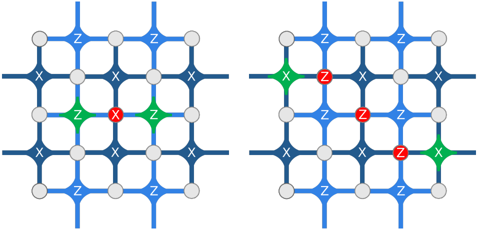




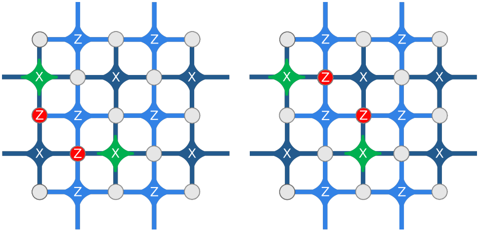
Syndrome



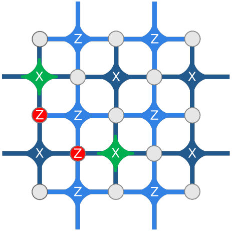
Syndrome

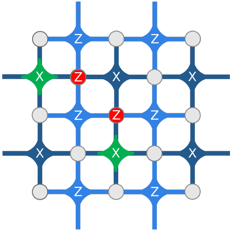


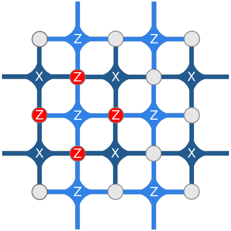
Benign ambiguity



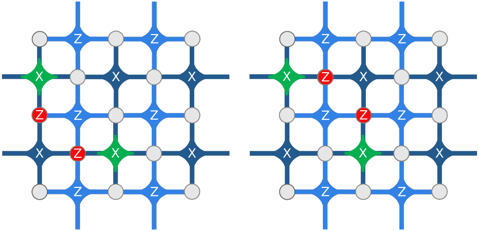
Benign ambiguity

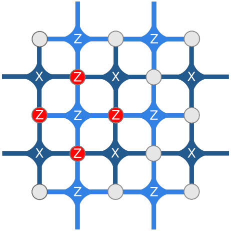






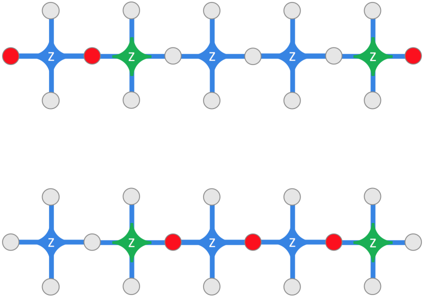
Benign ambiguity





$$Error = \prod_i \sigma_i^z = Z$$

Real ambiguity



Code distance

Minimal number of qubits required to form a logical error.

Code distance

Minimal number of qubits required to form a logical error.

i.e. system size.

Code distance

Minimal number of qubits required to form a logical error.

i.e. system size.

Larger code distance

- More redundancy
- Less logical error (assuming independent/local single physical qubit error)
- More processing power required

Code distance

Minimal number of qubits required to form a logical error.

i.e. system size.

Larger code distance

- More redundancy
- Less logical error (assuming independent/local single physical qubit error)
- More processing power required

Code distance

Minimal number of qubits required to form a logical error.

i.e. system size.

Larger code distance

- More redundancy
- Less logical error (assuming independent/local single physical qubit error)
- More processing power required

Code distance

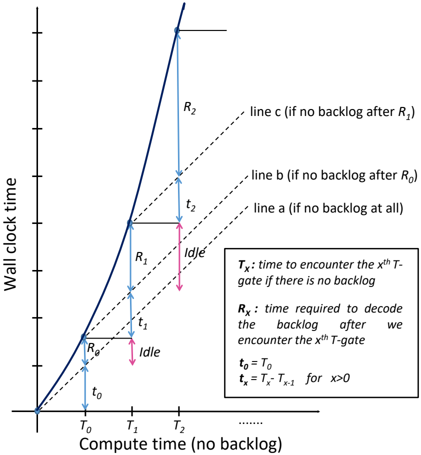
Minimal number of qubits required to form a logical error.

i.e. system size.

Larger code distance

- More redundancy
- Less logical error (assuming independent/local single physical qubit error)
- More processing power required

Scaling



Wall clock time

Compute time (no backlog)

T_0 T_1 T_2 ....

t_0 t_1 t_2

R_0 R_1 R_2

Idle

Idle

line c (if no backlog after R_1)

line b (if no backlog after R_0)

line a (if no backlog at all)

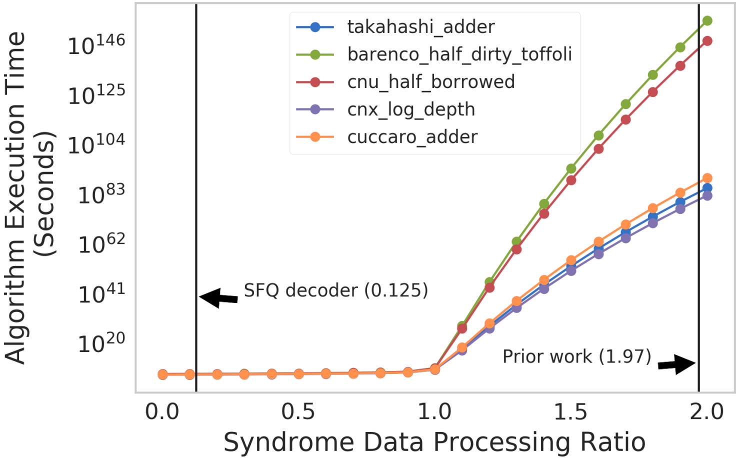
***T_x*:** *time to encounter the x^{th} T-gate if there is no backlog*

***R_x*:** *time required to decode the backlog after we encounter the x^{th} T-gate*

$t_0 = T_0$

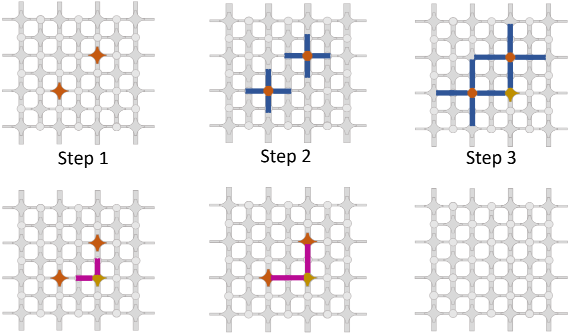
$t_x = T_x - T_{x-1}$ *for* $x>0$

Scaling



takahashi_adder
barenco_half_dirty_toffoli
cnu_half_borrowed
cnx_log_depth
cuccaro_adder
SFQ decoder (0.125)
Prior work (1.97)
Algorithm Execution Time (Seconds)
10^{146}
10^{125}
10^{104}
10^{83}
10^{62}
10^{41}
10^{20}
0.0
0.5
1.0
1.5
2.0
Syndrome Data Processing Ratio

Algorithm



Step 1

Step 2

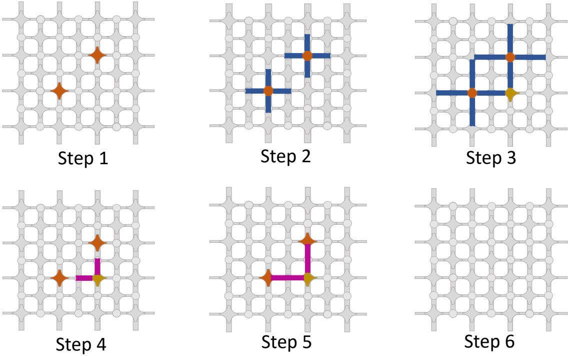
Step 3

Step 4

Step 5

Step 6

Algorithm



Step 1

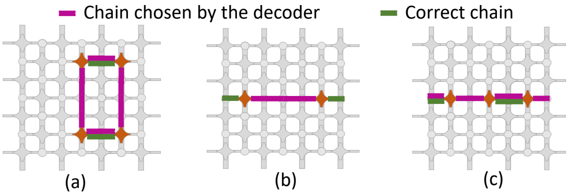
Step 2

Step 3

Step 4

Step 5

Step 6



(a)

(b)

(c)

Implementation and performance

- Hardware decoding
- Low power
- High speed

Implementation and performance

- Hardware decoding
- Low power
- High speed

Implementation and performance

- Hardware decoding
- Low power
- High speed

Implementation and performance

- Hardware decoding
- Low power
- High speed

Code Distance	Max (ns)	Average (ns)
3	3.74	0.28
5	9.28	0.72
7	14.2	2.00
9	19.2	3.81

Implementation and performance

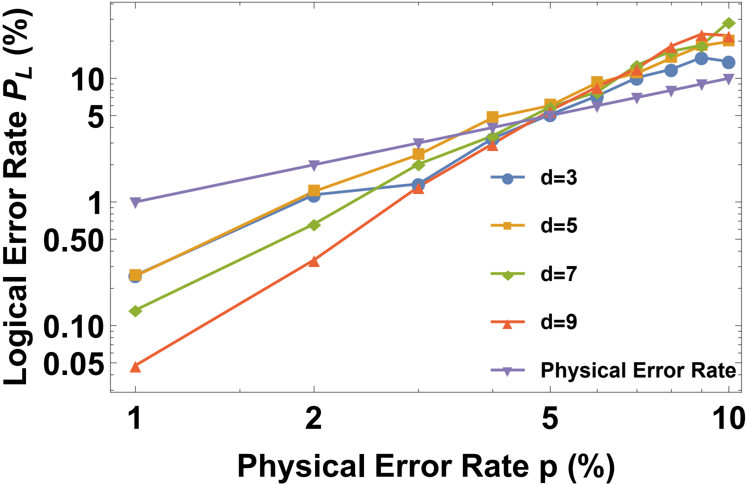
- Hardware decoding
- Low power
- High speed

Code Distance	Max (ns)	Average (ns)
3	3.74	0.28
5	9.28	0.72
7	14.2	2.00
9	19.2	3.81

Power concumption

3.78 mW for code distance 9.

Implementation and performance


Logical Error Rate P_L (%)
Physical Error Rate p (%)
d=3
d=5
d=7
d=9
Physical Error Rate