http://aqua.sfc.wide.ad.jp

Advances Toward Realizing the Surface Code

Rodney Van Meter, Keio University
http://aqua.sfc.wide.ad.jp/

FIRST Project Annual All-Hands Meeting @ Kyoto
2011 Dec 13-16

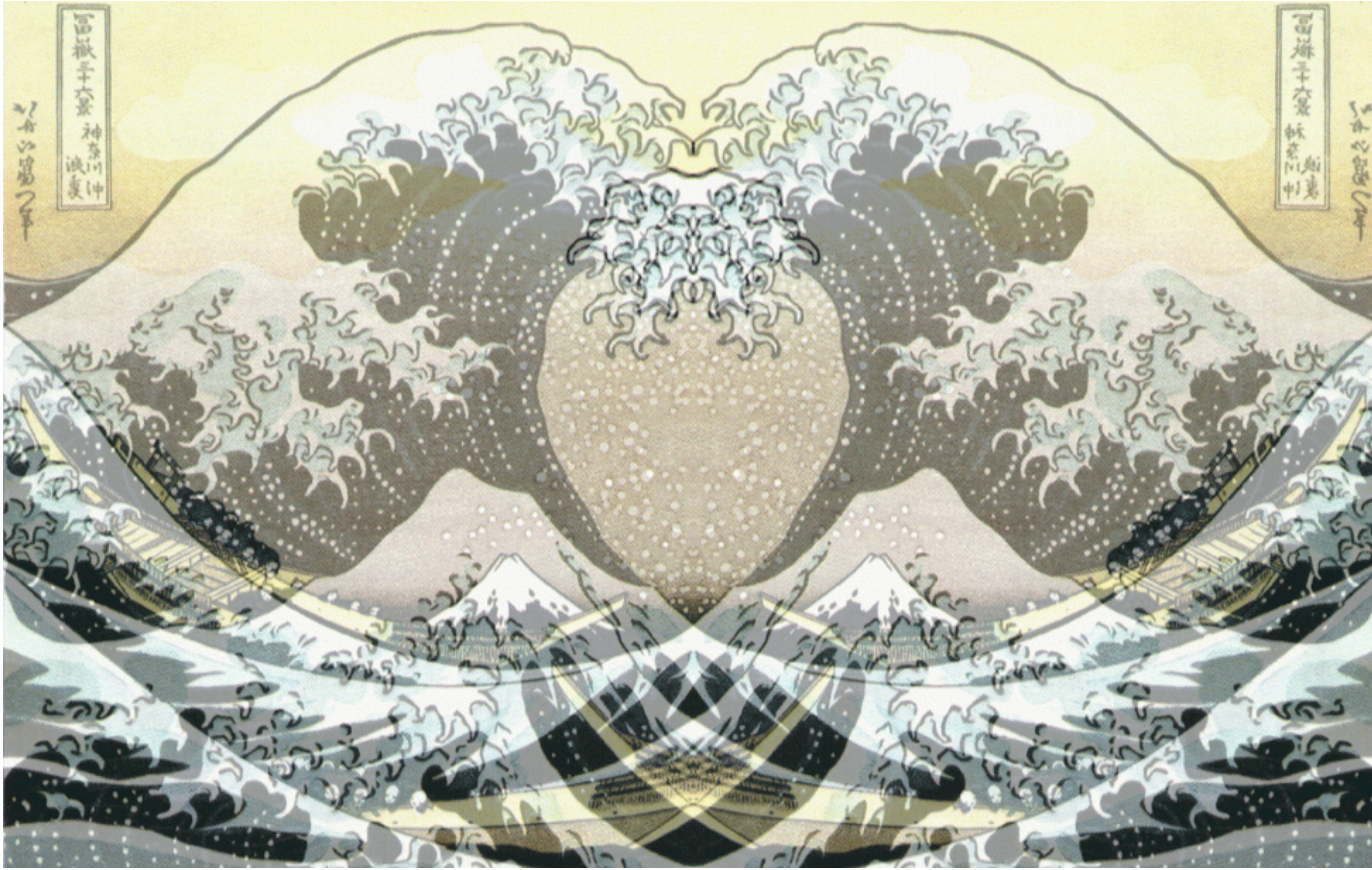
(all the good work, slides and animations are by Shota Nagayama, Austin Fowler, Clare Horsman, Cody Jones and my undergrads -- I just go surfing while they work)

Aqua & Friends

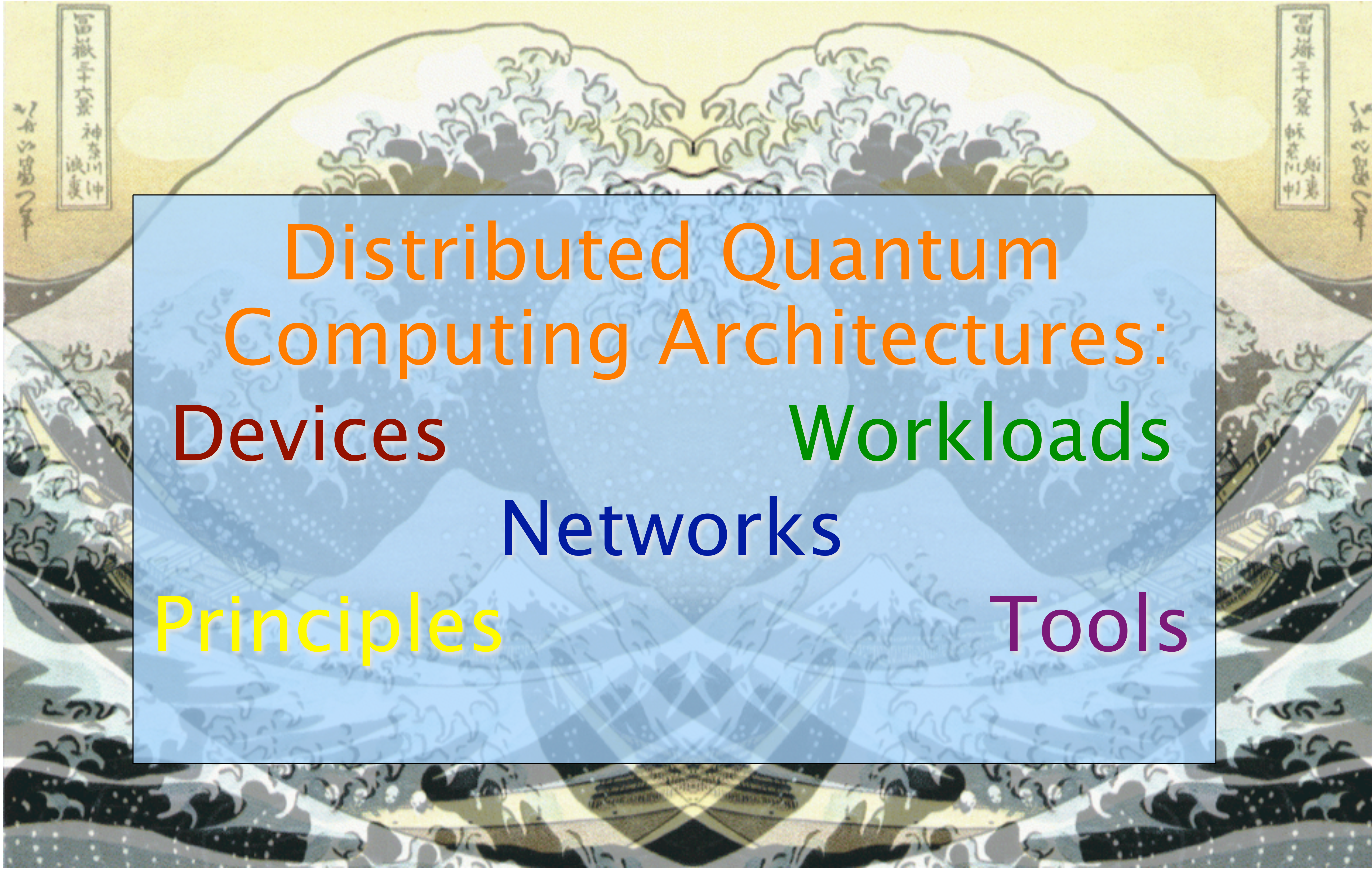
Aqua & Friends

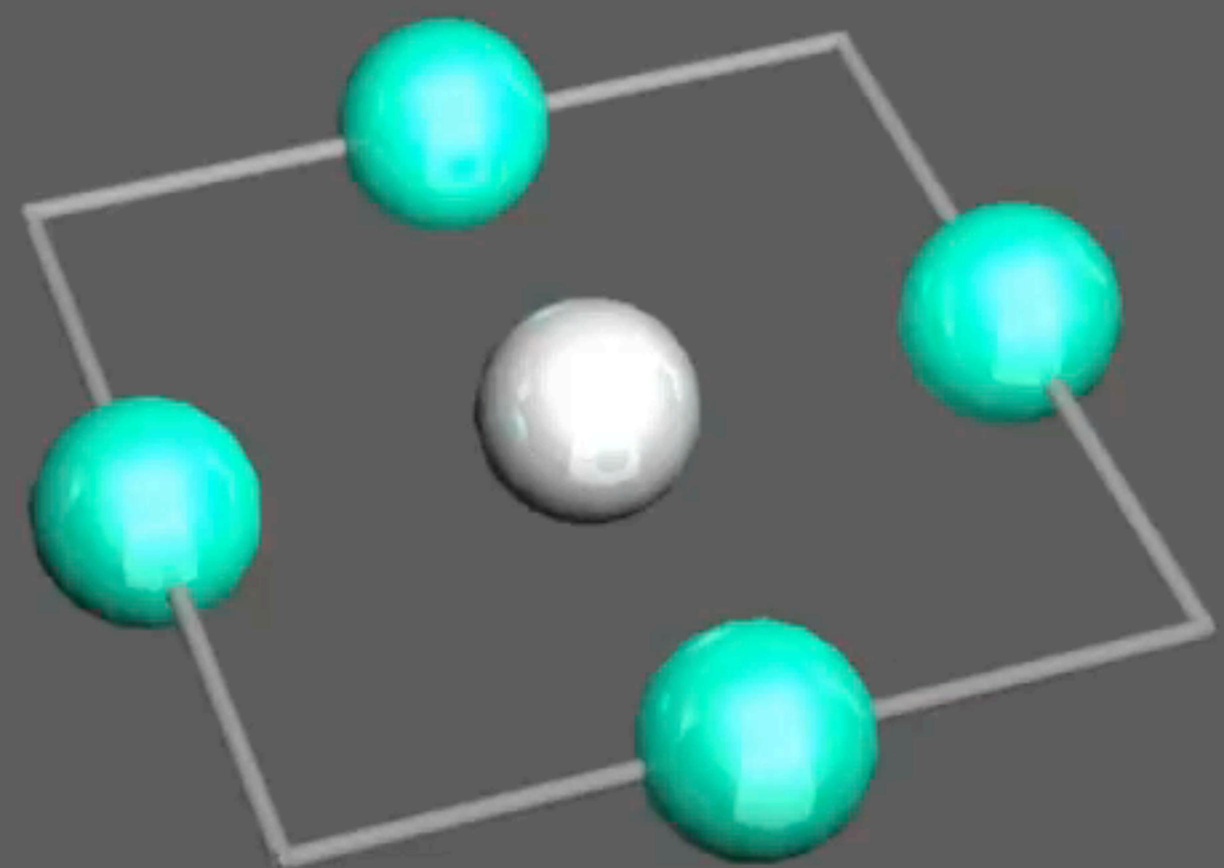

Google

AQUA: Advancing Quantum Architecture



AQUA: Advancing Quantum Architecture





5

Surface Code Strengths

Surface Code Strengths

- Simple, 2-D or 3-D nearest-neighbor-only operation (physical feasibility high!)

Surface Code Strengths

- Simple, 2-D or 3-D nearest-neighbor-only operation (physical feasibility high!)
- High threshold for gate, memory, meas. errors
  - ~1% for large systems (arXiv:1110.5133v1 [quant-ph]) (n.b.: slightly worse than previously said for large sys)
  - possibly 2-3% for small experiments

Surface Code Strengths

- Simple, 2-D or 3-D nearest-neighbor-only operation (physical feasibility high!)
- High threshold for gate, memory, meas. errors
  - ~1% for large systems (arXiv:1110.5133v1 [quant-ph]) (n.b.: slightly worse than previously said for large sys)
  - possibly 2-3% for small experiments
- Flexible:
  - strength of EC grows incrementally (compare to concatenated CSS codes)
  - software-assigned resources
  - easy movement of logical qubits

Surface Code Strengths

- Simple, 2-D or 3-D nearest-neighbor-only operation (physical feasibility high!)
- High threshold for gate, memory, meas. errors
  - ~1% for large systems (arXiv:1110.5133v1 [quant-ph]) (n.b.: slightly worse than previously said for large sys)
  - possibly 2-3% for small experiments
- Flexible:
  - strength of EC grows incrementally (compare to concatenated CSS codes)
  - software-assigned resources
  - easy movement of logical qubits
- Supporting classical processing achievable

Scalability: fault-tolerance

- Trade-off between resources and threshold
- Thresholds
  - unlimited range, unlimited qubits: ~ 10^{-2}

    Knill, quant-ph/0410199
  - unlimited range, many qubits: ~ 10^{-3}–10^{-4}

    Steane, Phys. Rev. A 68, 042322 (2003)
  - 2D lattice, nearest neighbor: ~ 10^{-5}

    Svore, QIC 7, 297 (2007)
  - bilinear nearest neighbor: ~ 10^{-6}

    Stephens, QIC 8, 330 (2008)
  - linear nearest neighbor: ~ 10^{-5}

    Stephens, PRA 80, 022313 (2009)

Space Overhead Example: "Racetrack" Architecture

6,000,000,000 imperfect qubits

12,000 perfect qubits
for Shor's algorithm for 2,048 bits

rdv *et al., IJQI* 8, 295 (2010) arXiv:0906.2686v2 [quant-ph]

Space Overhead Example: "Racetrack" Architecture

6,000,000,000 imperfect qubits

RHG Surface code w/ d = 56

gate error rate 0.2%, *KQ* ~ 10^15

12,000 perfect qubits

for Shor's algorithm for 2,048 bits

rdv *et al., IJQI* 8, 295 (2010) arXiv: 0906.2686v2 [quant-ph]

Space Overhead Example: "Racetrack" Architecture

6,000,000,000 imperfect qubits

x8

Singular state "factories" for non-Clifford gates

x10,000

RHG Surface code w/ d = 56
gate error rate 0.2%, *KQ* ~ 10^15

12,000 perfect qubits
for Shor's algorithm for 2,048 bits

rdv *et al., IJQI* 8, 295 (2010) arXiv:0906.2686v2 [quant-ph]

Space Overhead Example: “Racetrack” Architecture

6,000,000,000 imperfect qubits

x1.25 — “Wiring” space to move qubits around

x8 — Singular state “factories” for non-Clifford gates

x10,000 — RHG Surface code w/ d = 56
gate error rate 0.2%, *KQ* ~ 10^15

12,000 perfect qubits
for Shor’s algorithm for 2,048 bits

rdv *et al., IJQI* 8, 295 (2010) arXiv:0906.2686v2 [quant-ph]

Space Overhead Example: "Racetrack" Architecture

6,000,000,000 imperfect qubits

x5 — Overhead to work around assumed 40% yield

x1.25 — "Wiring" space to move qubits around

x8 — Singular state "factories" for non-Clifford gates

x10,000 — RHG Surface code w/ d = 56
gate error rate 0.2%, *KQ* ~ 10^15

12,000 perfect qubits
for Shor's algorithm for 2,048 bits

rdv *et al.*, *IJQI* 8, 295 (2010) arXiv:0906.2686v2 [quant-ph]

Space Overhead Example: "Racetrack" Architecture

6,000,000,000 imperfect qubits

x10,000

RHG Surface code w/ d = 56

gate error rate 0.2%, *KQ* ~ 10^15

12,000 perfect qubits

for Shor's algorithm for 2,048 bits

rdv *et al., IJQI* 8, 295 (2010) [arXiv: 0906.2686v2](arXiv:0906.2686v2) [quant-ph]

Space Overhead Example: “Racetrack” Architecture

6,000,000,000 imperfect qubits

Gate & memory error rate have huge impact here

x10,000

RHG Surface code w/ d = 56
gate error rate 0.2%, *KQ* ~ 10^15

12,000 perfect qubits
for Shor’s algorithm for 2,048 bits

rdv *et al., IJQI* 8, 295 (2010) arXiv:0906.2686v2 [quant-ph]

Space Overhead Example: "Racetrack" Architecture

6,000,000,000 imperfect qubits

Singular state "factories" for non-Clifford gates

12,000 perfect qubits
for Shor's algorithm for 2,048 bits

rdv *et al., IJQI* 8, 295 (2010) arXiv:0906.2686v2 [quant-ph]

Space Overhead Example: "Racetrack" Architecture

6,000,000,000 imperfect qubits

Direct tradeoff of
speed v. space here:
1/8 the space, 1/8 the speed

x8

Singular state "factories" for non-Clifford gates

12,000 perfect qubits
for Shor's algorithm for 2,048 bits

rdv *et al., IJQI* 8, 295 (2010) [arXiv: 0906.2686v2] [quant-ph]

Space Overhead Example: "Racetrack" Architecture

6,000,000,000 imperfect qubits

"Wiring" space to move qubits around

12,000 perfect qubits
for Shor's algorithm for 2,048 bits

rdv *et al., IJQI* 8, 295 (2010) arXiv: 0906.2686v2 [quant-ph]

Space Overhead Example: “Racetrack” Architecture

6,000,000,000 imperfect qubits

x1.25

“Wiring” space to move qubits around

Not too bad

12,000 perfect qubits
for Shor’s algorithm for 2,048 bits

rdv *et al., IJQI* 8, 295 (2010) arXiv:0906.2686v2 [quant-ph]

Space Overhead Example: "Racetrack" Architecture

6,000,000,000 imperfect qubits

Overhead to work around assumed 40% yield

12,000 perfect qubits
for Shor's algorithm for 2,048 bits

rdv *et al., IJQI* 8, 295 (2010) arXiv: 0906.2686v2 [quant-ph]

Space Overhead Example: "Racetrack" Architecture

6,000,000,000 imperfect qubits

Overhead to work around assumed 40% yield

Looks bad, but analysis shows using stringent definition of "good" device keeps QEC overhead low

12,000 perfect qubits
for Shor's algorithm for 2,048 bits

rdv *et al.*, *IJQI* 8, 295 (2010) arXiv: 0906.2686v2 [quant-ph]

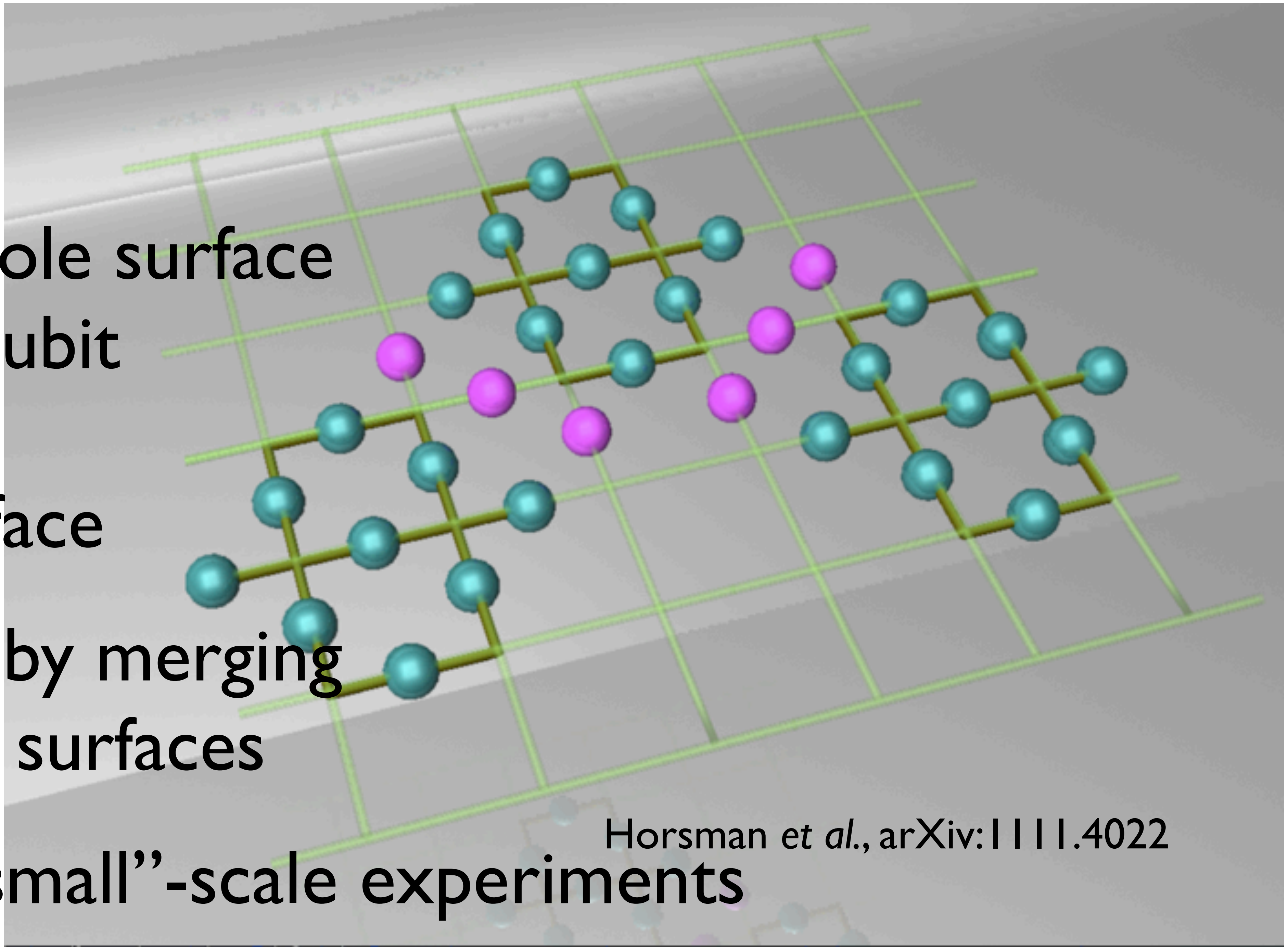
Aqua : Advancing Quantum Architecture

Outline/Results

- QuDOS architecture arXiv:1010.5022 [quant-ph]
- Lattice surgery for the planar code
  Horsman *et al.,* arXiv:1111.4022 [quant-ph]
- Surface code on a defective lattice
  Nagayama *et al.,* in preparation (poster tonight)
- Quantum picturalism
  Horsman, *NJP* 13, 095011 (2011)
- Graph embedding
  Choi & rdv, *ACM J. Emerging Tech. in Comp. Sys.* 7, 11 (2011)
- 2-D adder circuit
  Choi & rdv, *ACM JETC*, to appear (arXiv 1008.5093)
- Networking/repeater protocol development
  Aparicio (Best Student Paper Award @AINTEC 2011), multiplexing, Recursive Network Architecture, quantum Dijkstra for request routing, etc.
- Education & outreach

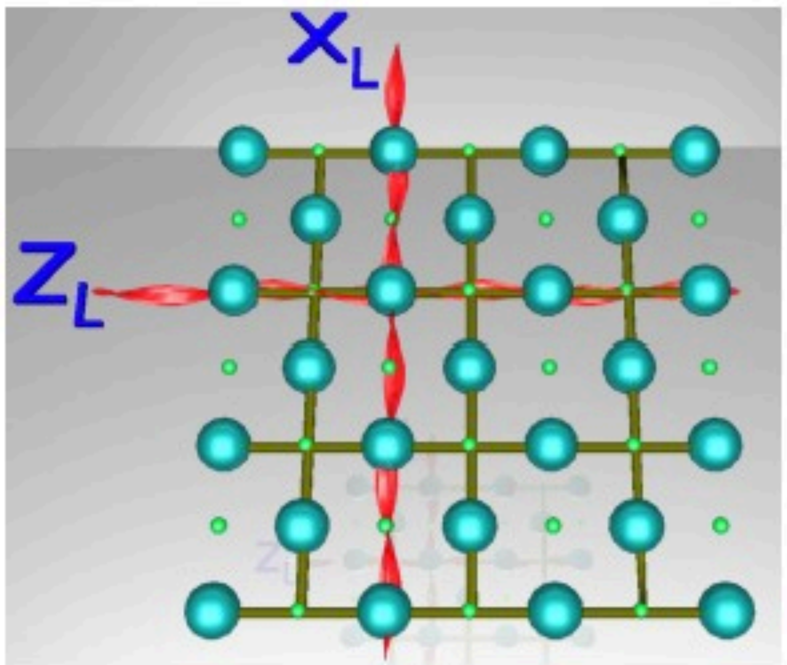
10

Lattice Surgery

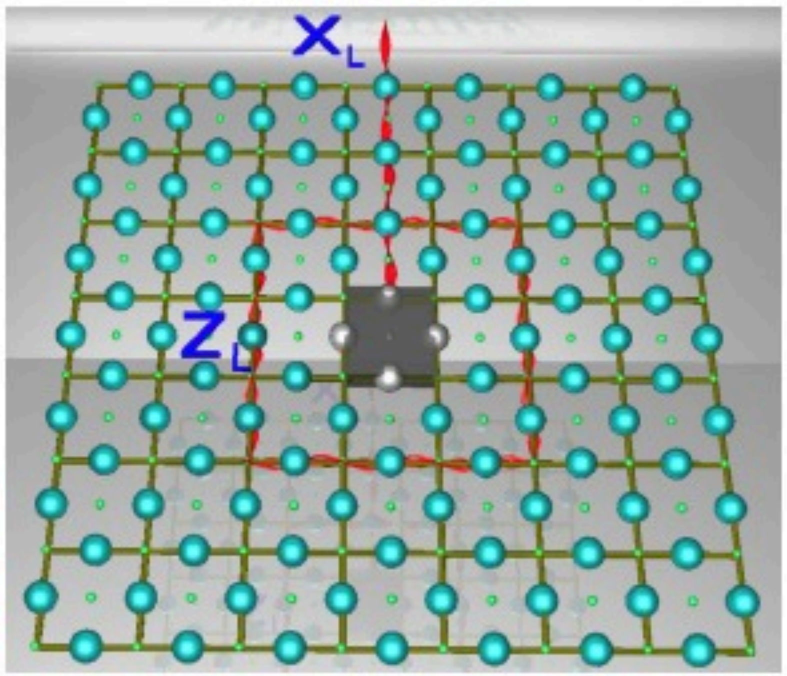


- Use one whole surface per logical qubit instead of holes in surface
- Gates done by merging and splitting surfaces
- Useful for "small"-scale experiments
- 53 qubits for distance-3 CNOT

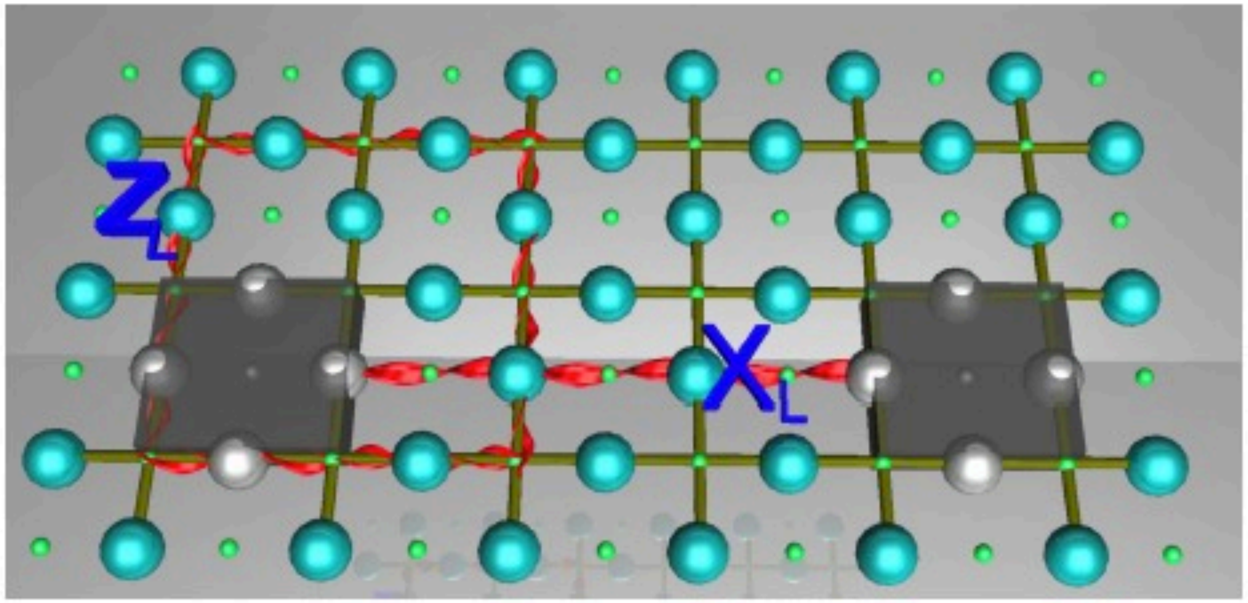
Three Forms of Surface Code



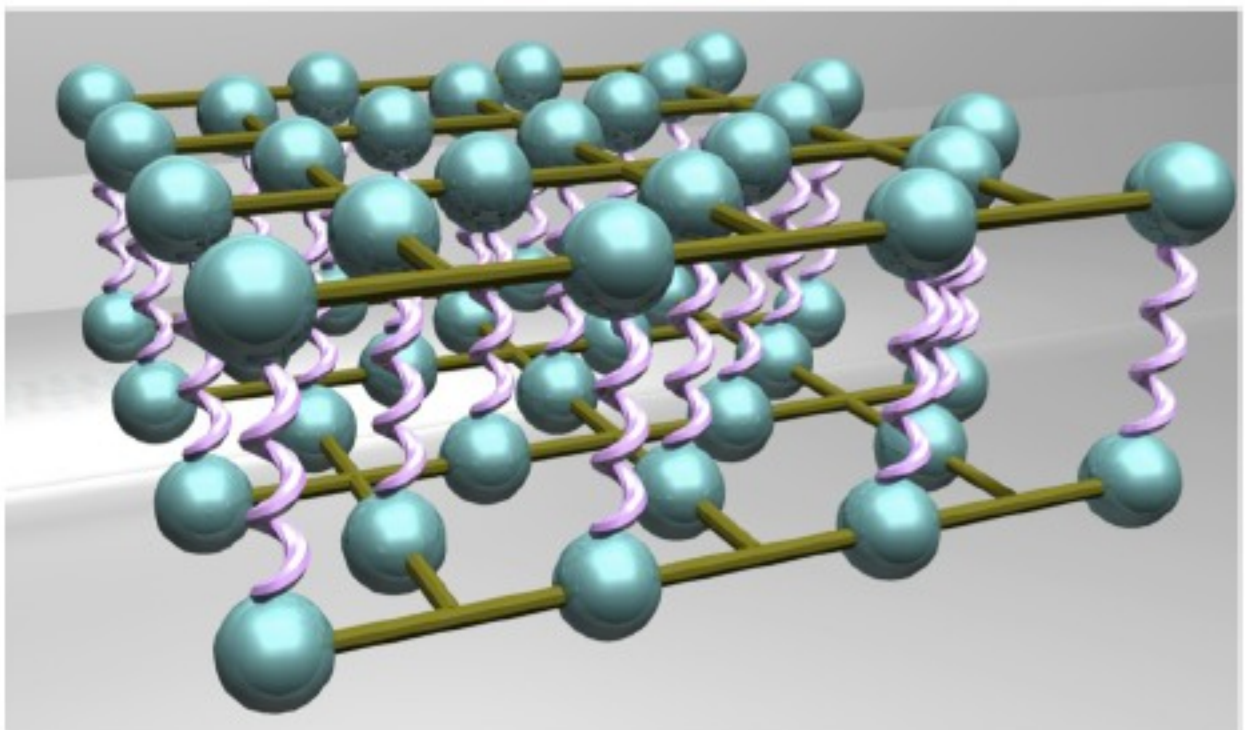
(a)



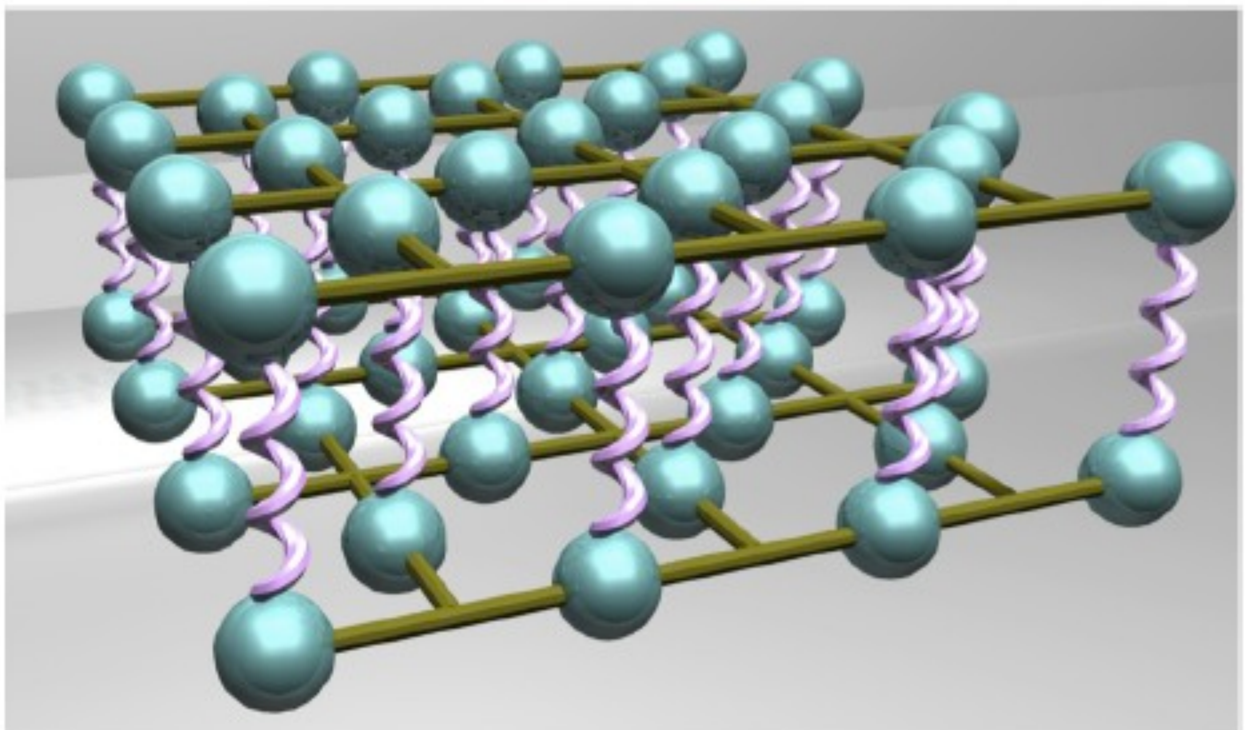
(b)

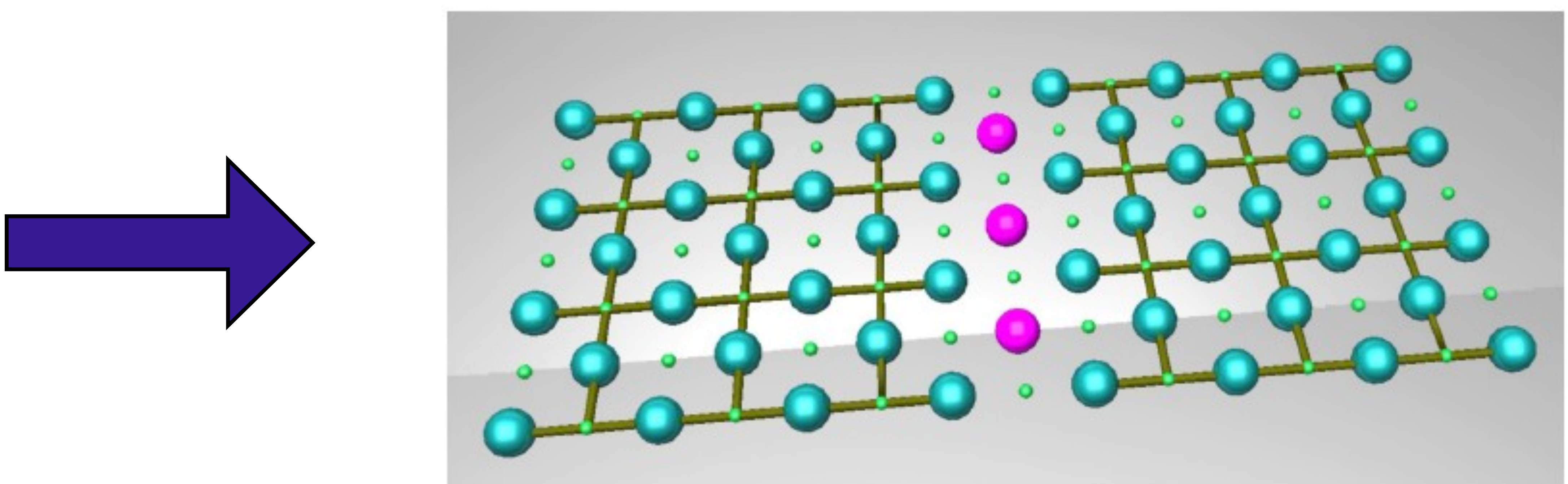


Transversal ==> Surgery



Transversal ==> Surgery





Merging and Splitting

- Merge operator takes two qubits to one

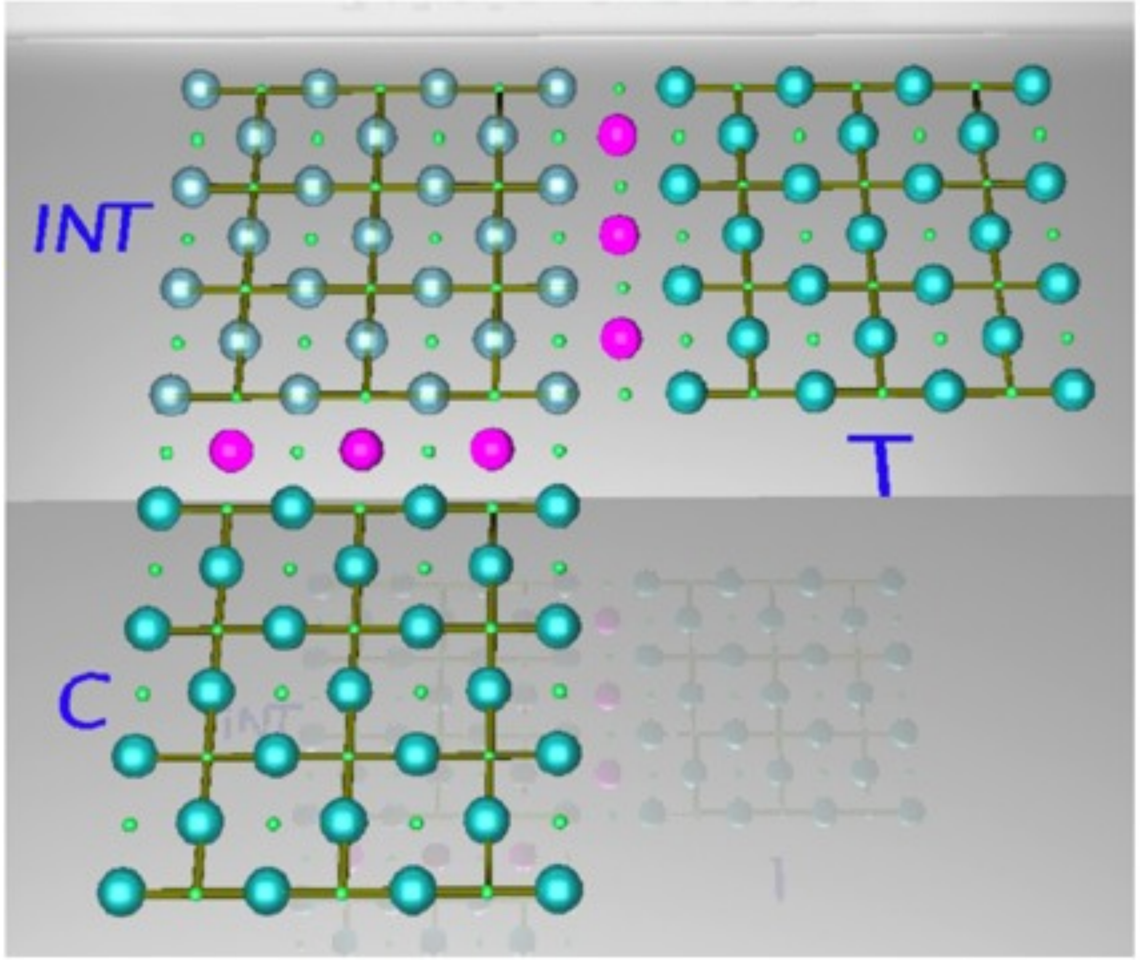
$$|\psi\rangle = \alpha|0\rangle_L + \beta|1\rangle_L$$

$$|\phi\rangle = \alpha'|0\rangle_L + \beta'|1\rangle_L$$

$$|\psi\rangle \circledM |\phi\rangle = \alpha|\phi\rangle + (-1)^M \beta|\bar{\phi}\rangle$$
$$= \alpha'|\psi\rangle + (-1)^M \beta'|\bar{\psi}\rangle$$

- Splitting surface turns one qubit into Bell pair-like state

$$\alpha|0\rangle_L + \beta|1\rangle_L \longrightarrow \alpha|00\rangle_L + \beta|11\rangle_L$$



CNOT

$$|C\rangle = \alpha|0\rangle_L + \beta|1\rangle_L$$

$$|T\rangle = \alpha'|0\rangle_L + \beta'|1\rangle_L$$

$$|INT\rangle = |0\rangle_L$$

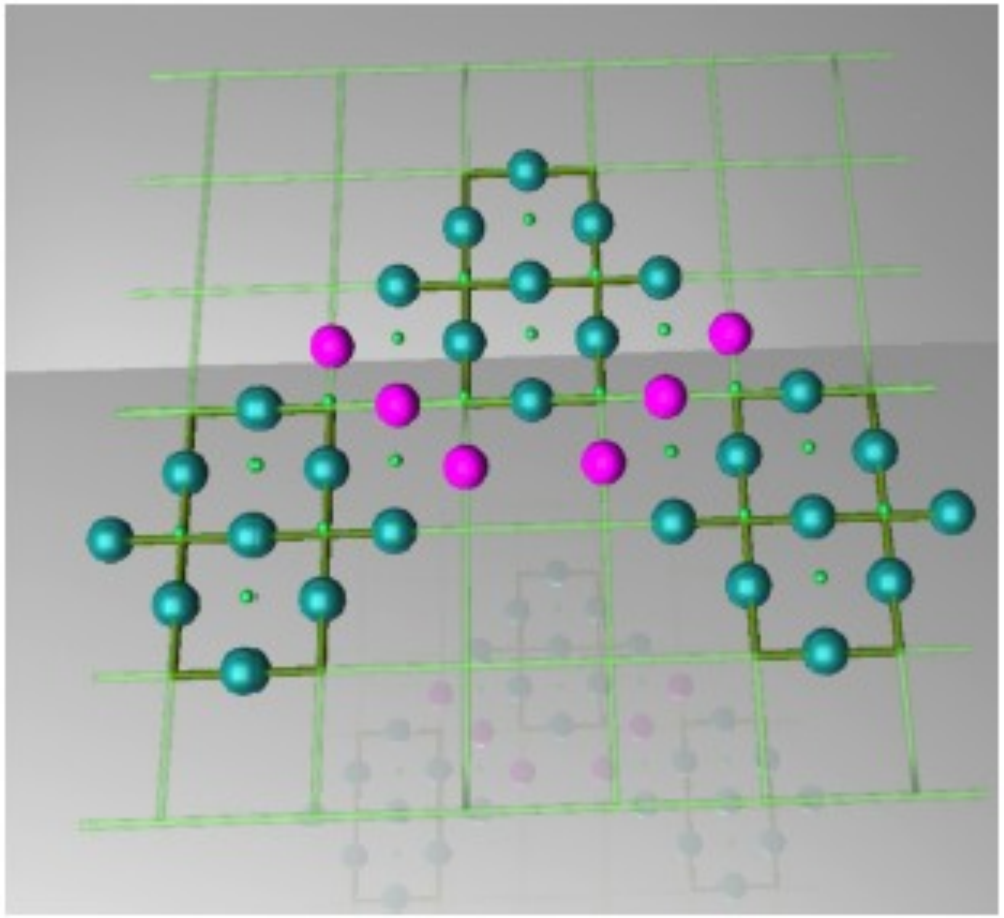
$$|C\rangle \textcircled{M} |INT\rangle = |C\rangle = \alpha|0\rangle_L + (-1)^M\beta|1\rangle_L$$

$$C' \; INT'\rangle = \alpha|00\rangle_L + (-1)^M\beta|11\rangle_L$$

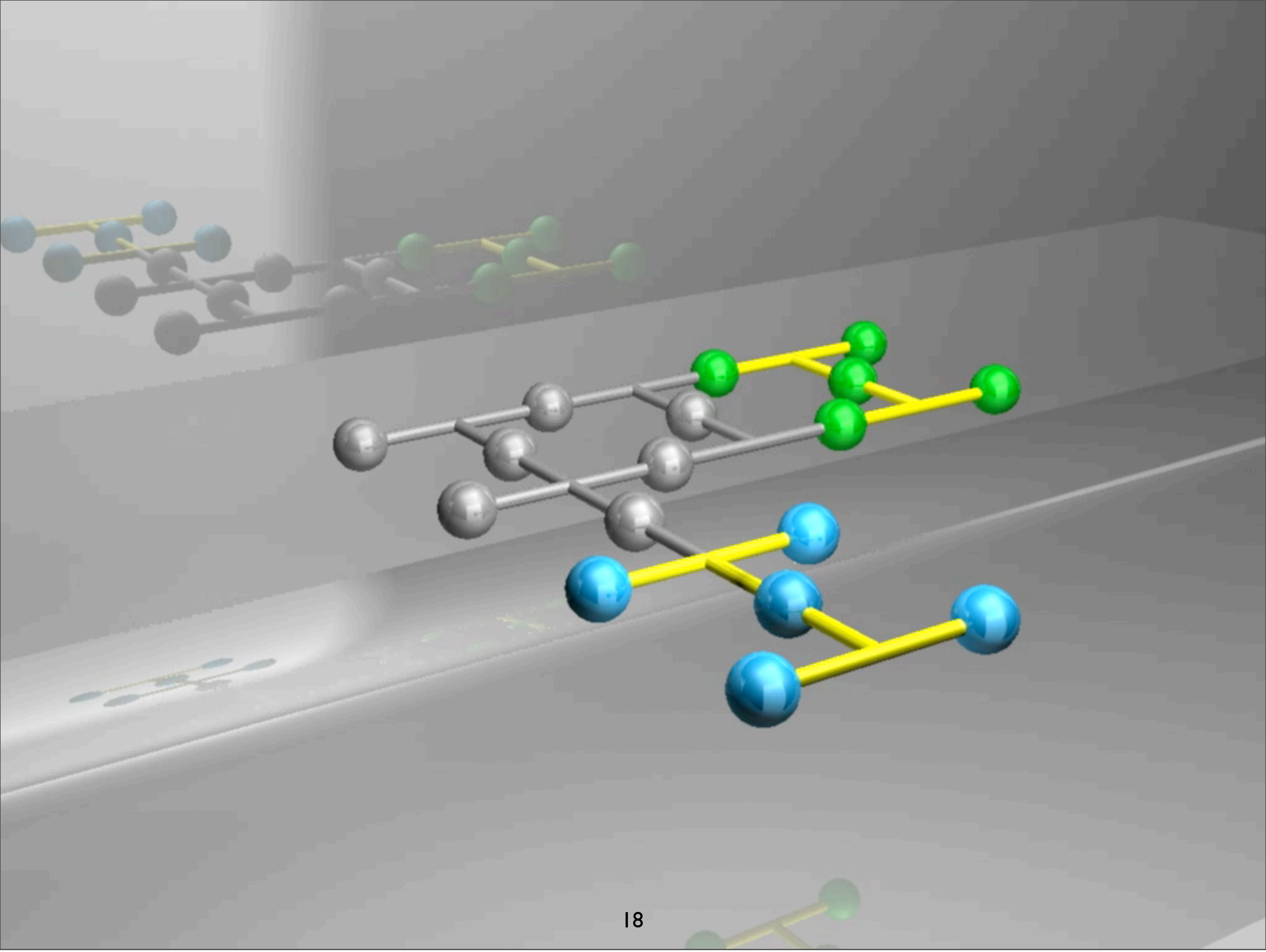
$$|C' \; (INT' \textcircled{M} T)\rangle = \alpha|0\rangle_L \otimes (|0\rangle_L \textcircled{M} |T\rangle) + (-1)^M\beta|1\rangle_L \otimes (|1\rangle_L \textcircled{M} |T\rangle)$$

$$= \alpha|0\rangle_L \otimes |T\rangle + (-1)^{(M+M')}\beta|1\rangle_L \otimes |\bar{T}\rangle$$

Proposed Experiments



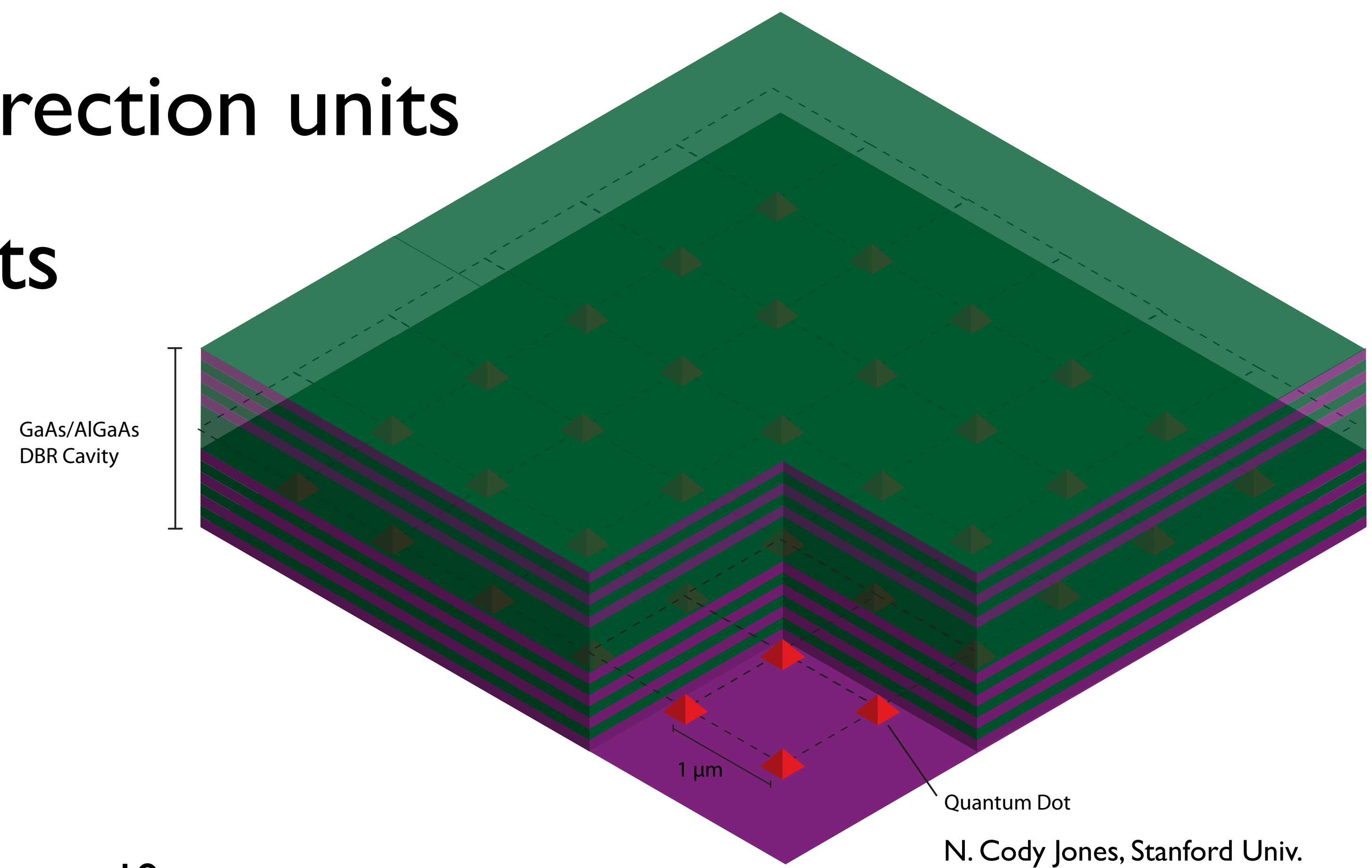
- 53 qubits for distance-3 CNOT
  - 33 data, 20 syndrome
  - 9+4 would be enough for single logical qubit
  - 21+12 enough for Bell pair creation
- Details on:
  - State injection
  - Hadamard gate
  - Movement of qubits
  - Behavior of “rough” v. “smooth” operations
- are in the paper arXiv:1111.4022 [quant-ph]



SC on a Defective Lattice

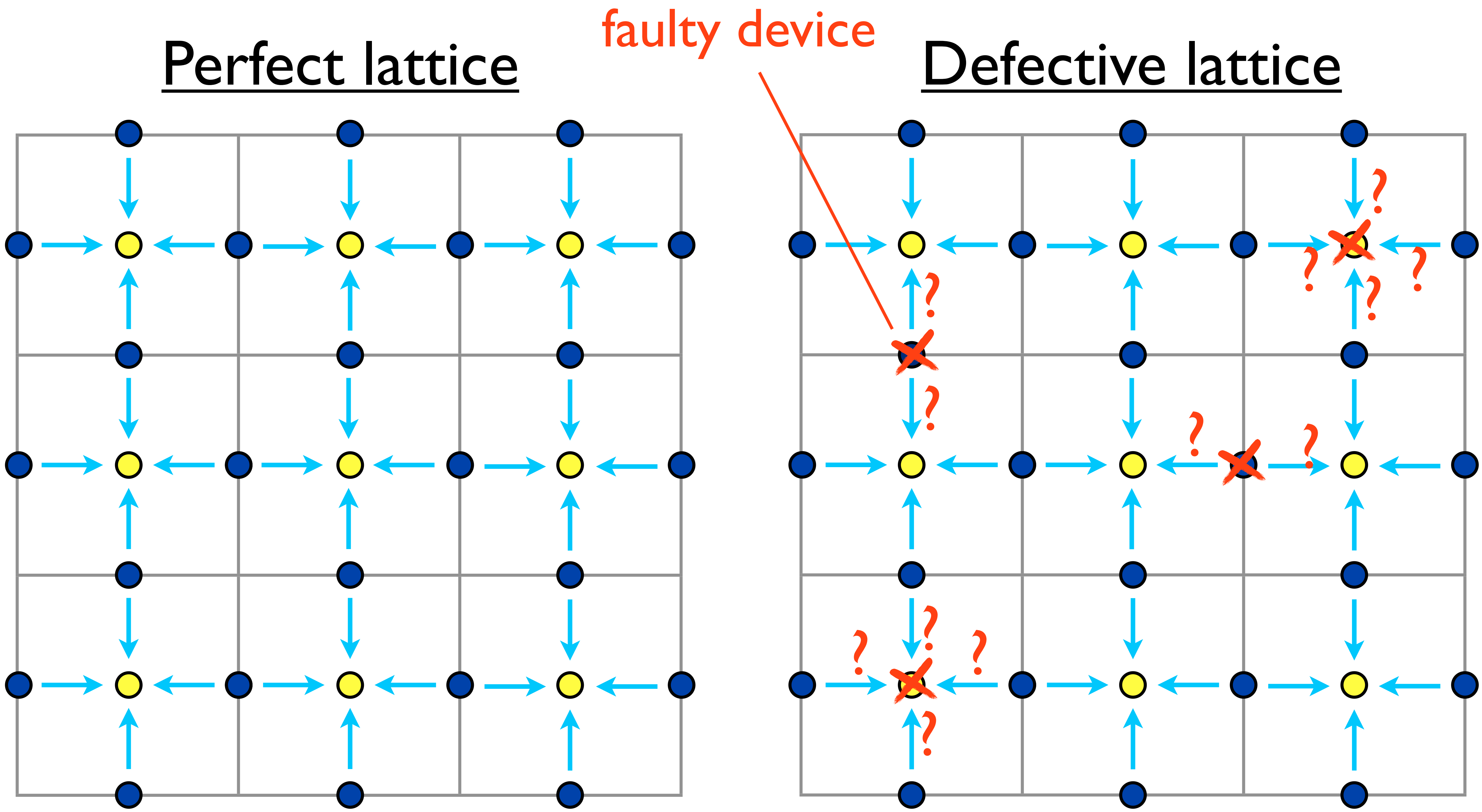
- Purpose
  - to develop surface code extension which deals with faulty devices

    ➡ to adapt surface code to realistic situation!
- Approach
  - modifying error correction units
  - scheduling QEC units
- Evaluation
  - threshold and scheduling/timing



N. Cody Jones, Stanford Univ.

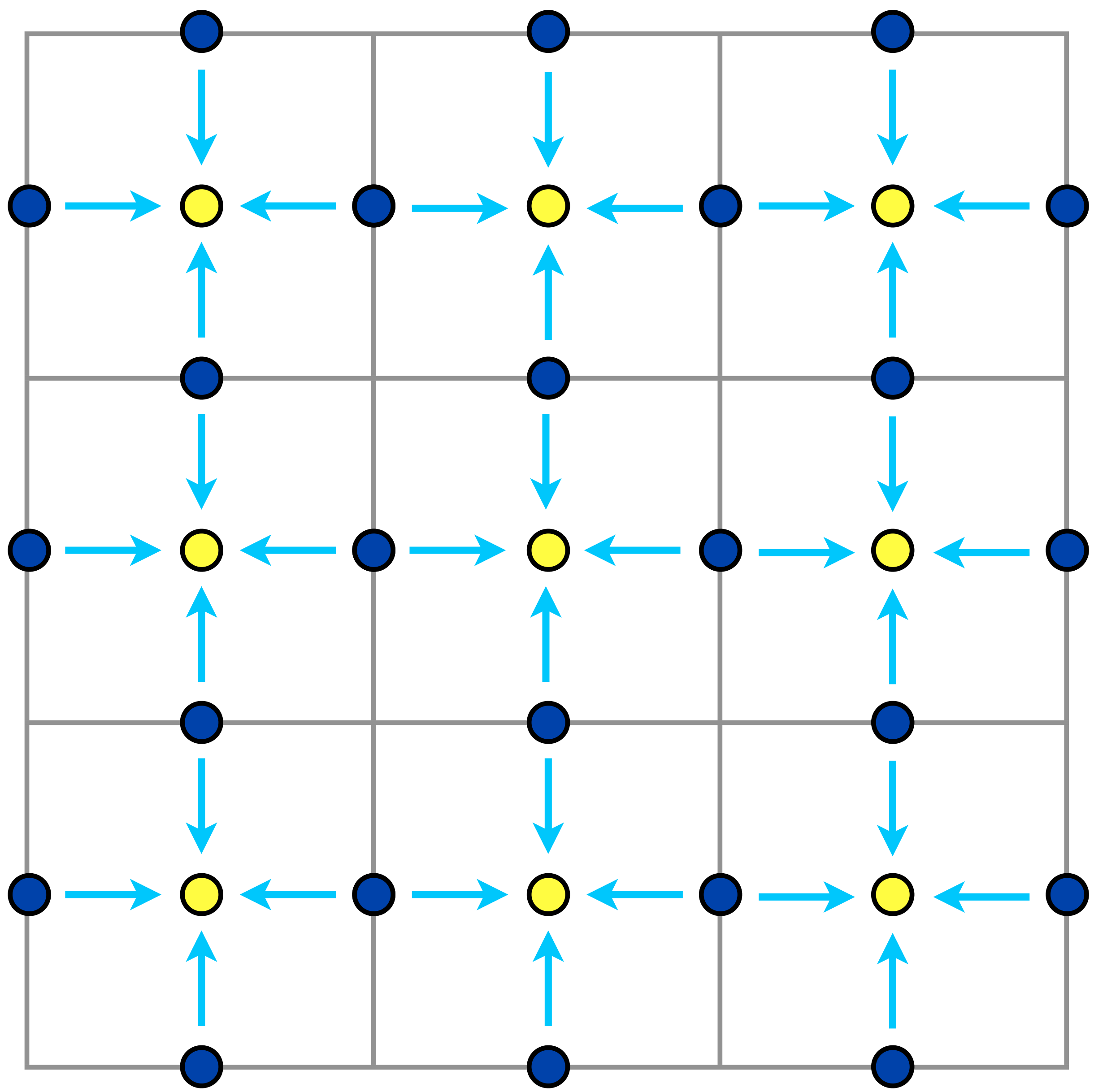
Defective lattice



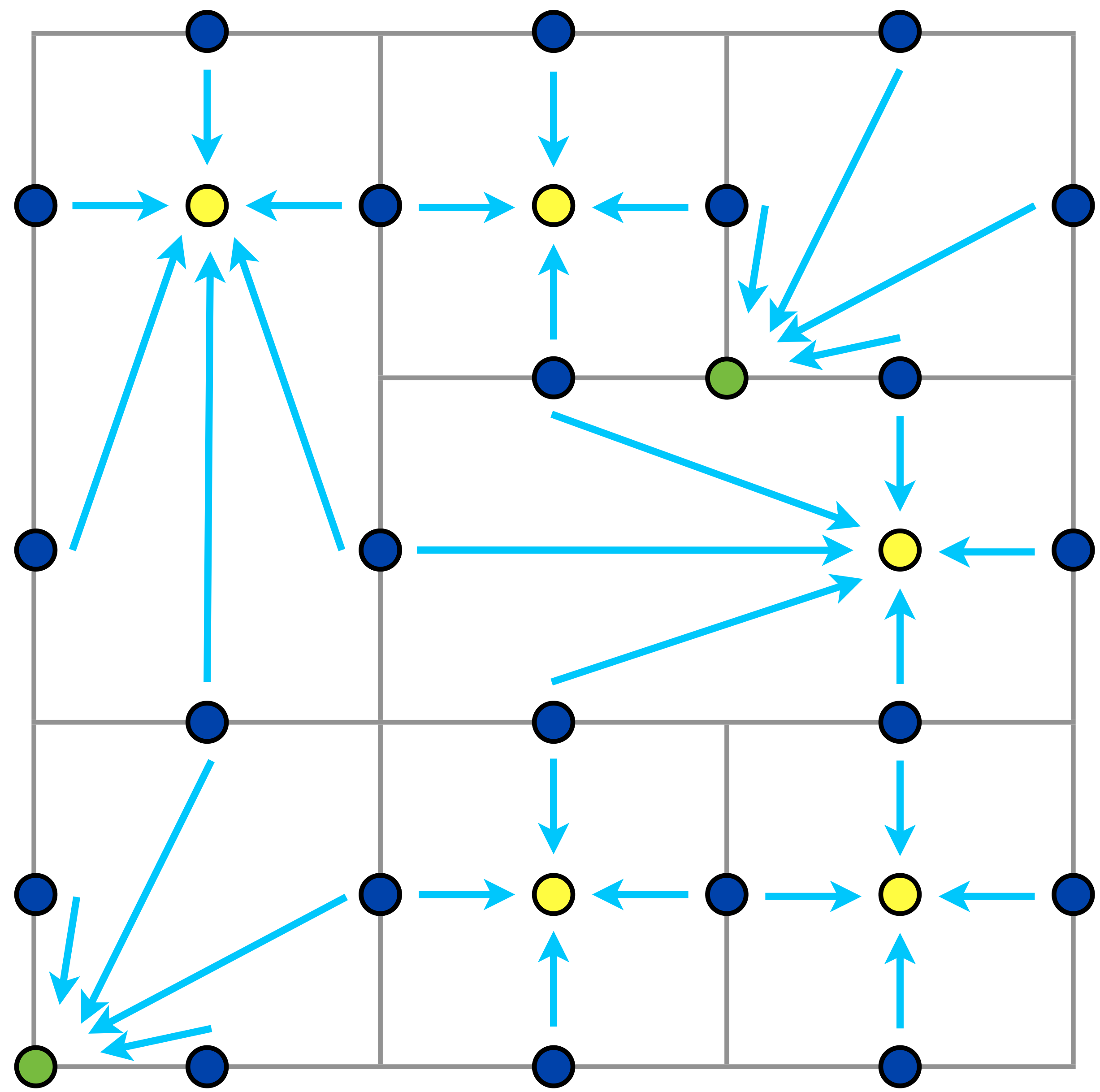
this is Z syndrome only

Idea

Perfect lattice



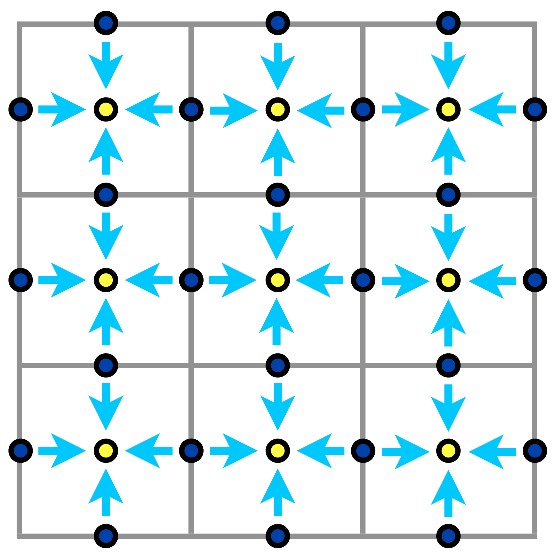
Defective lattice

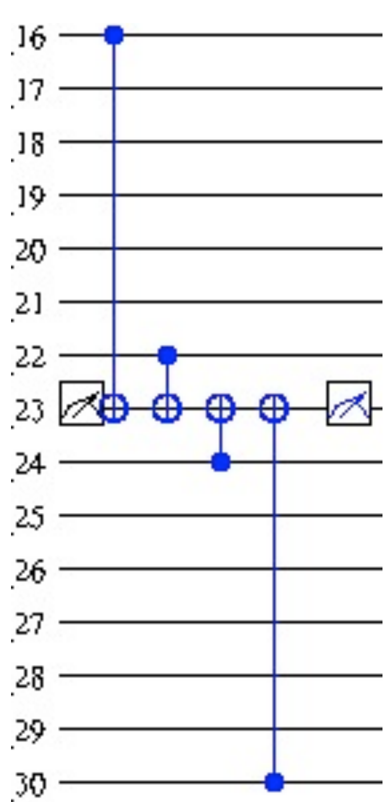


this is Z syndrome only

Idea

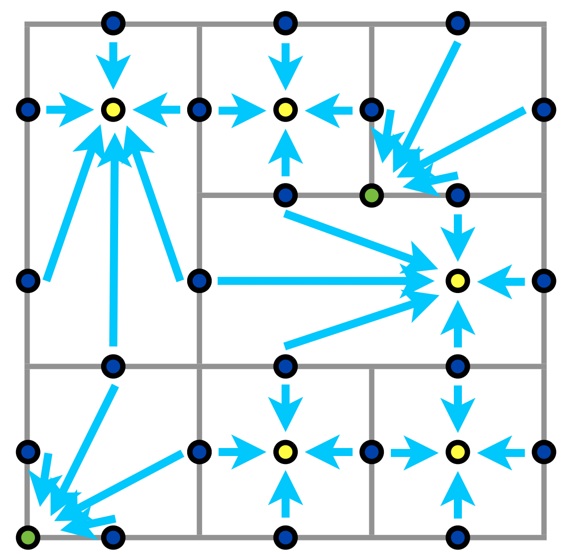
Perfect lattice

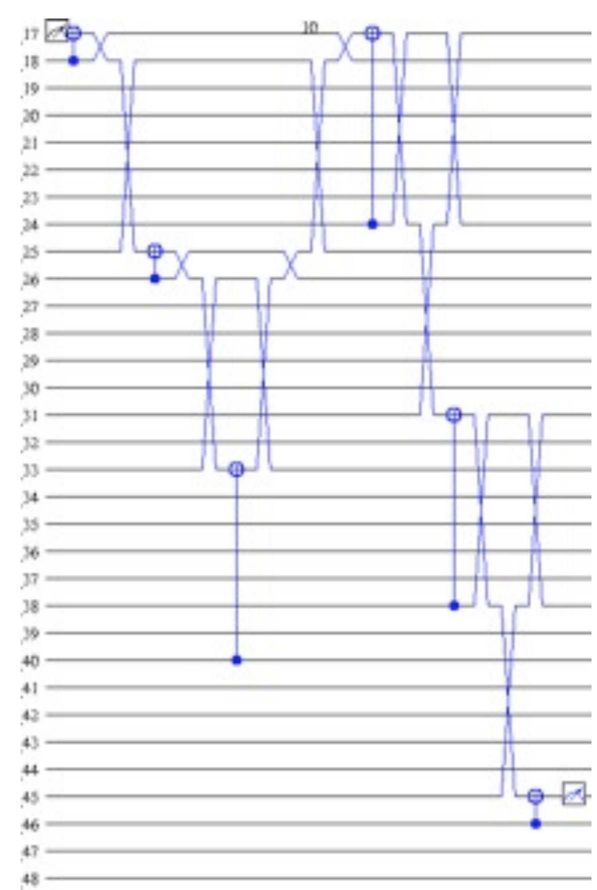




6 steps

Defective lattice

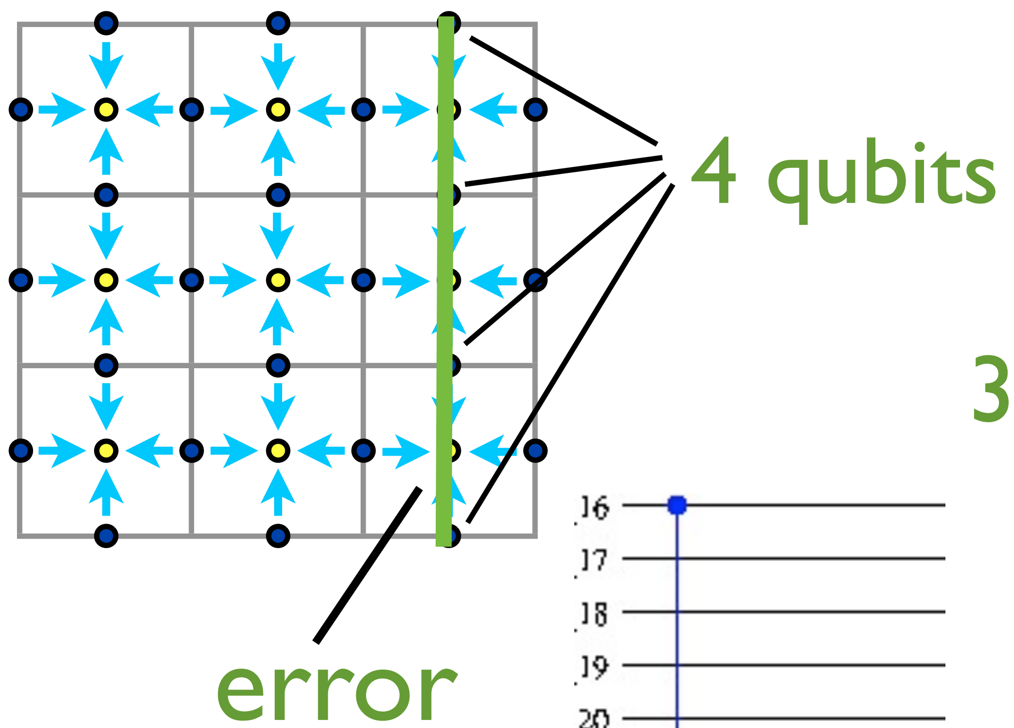


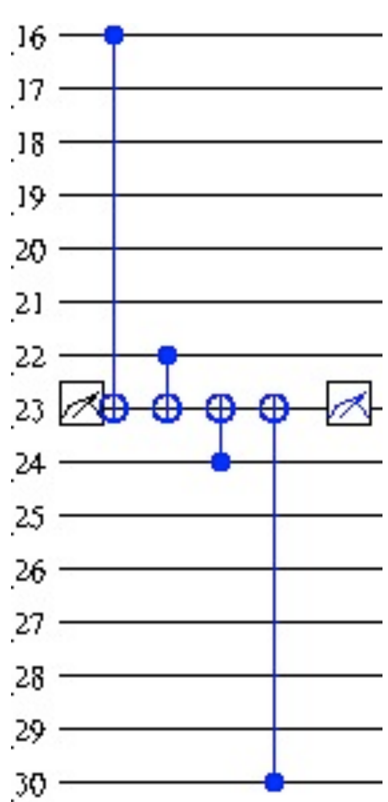


19 steps

Idea

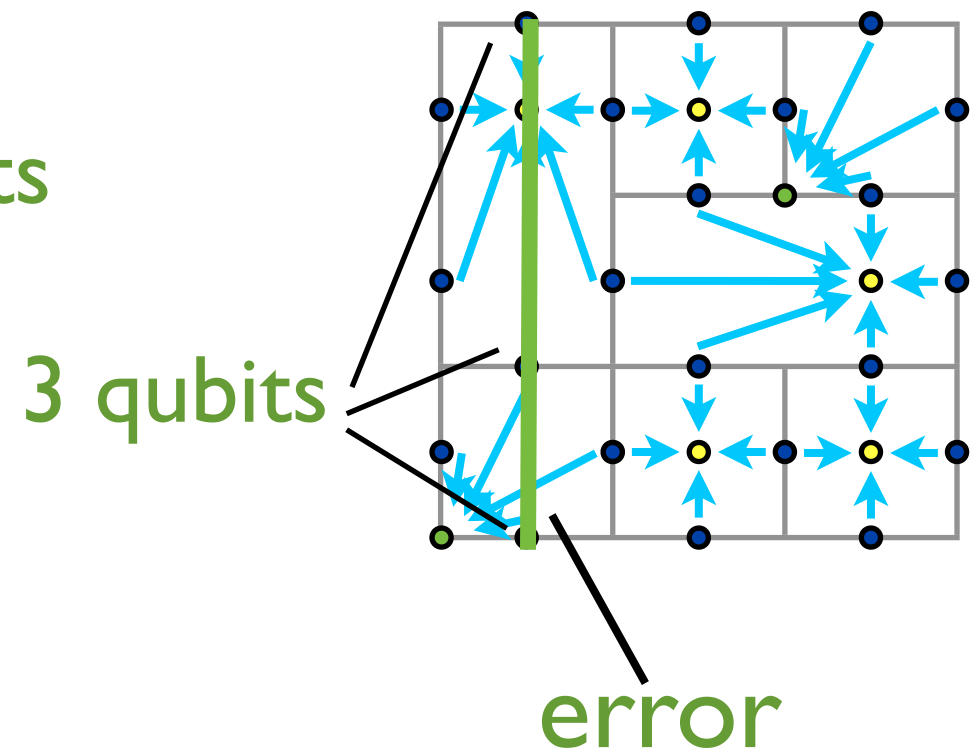
Perfect lattice

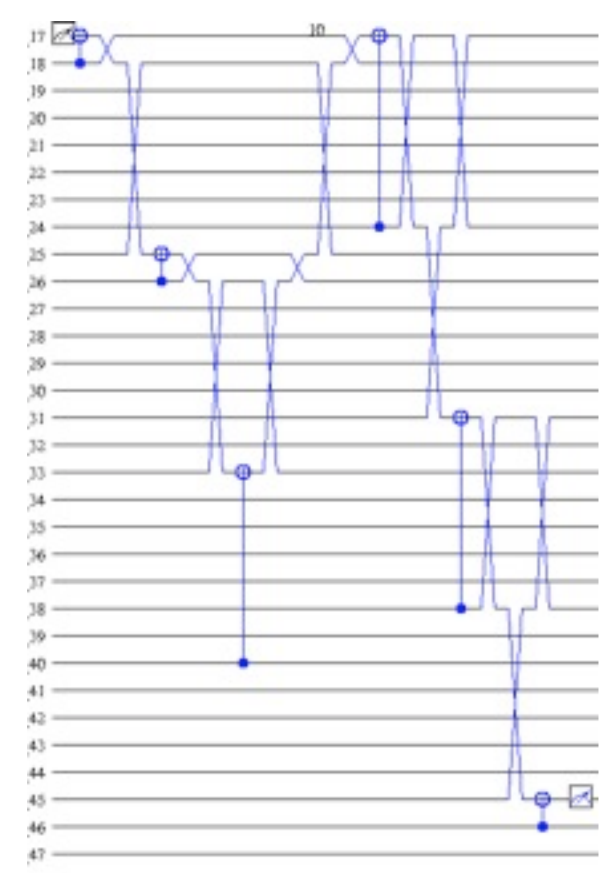




6 steps

Defective lattice

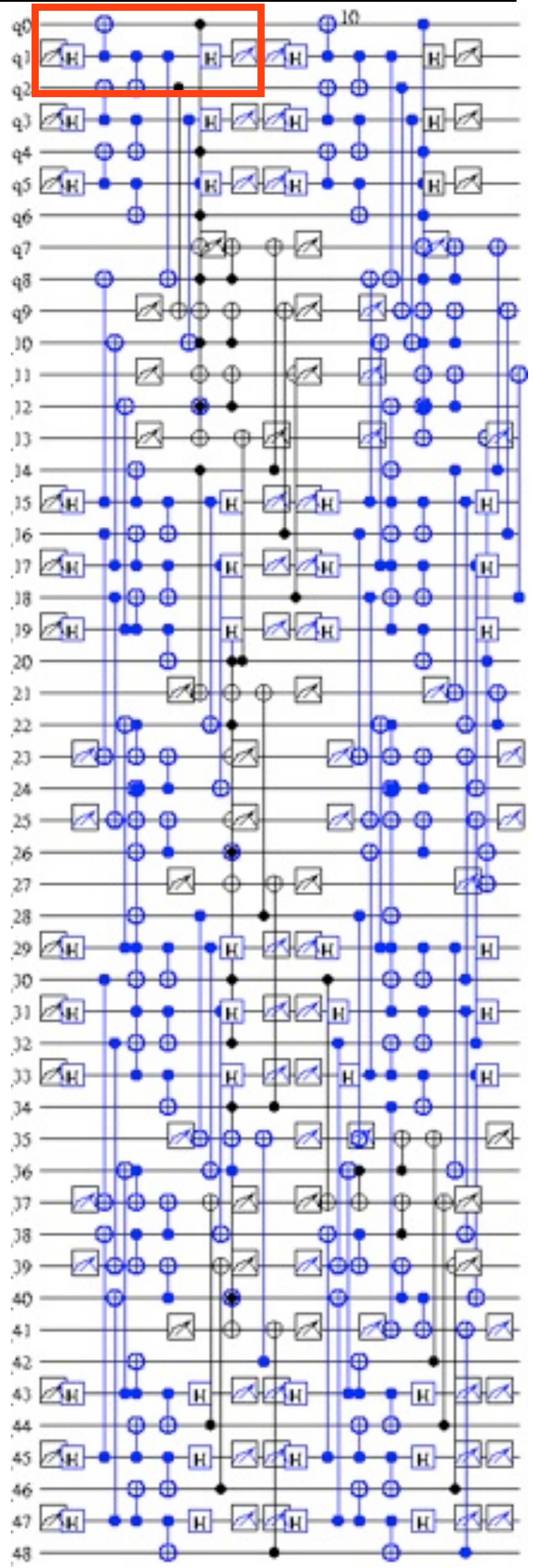




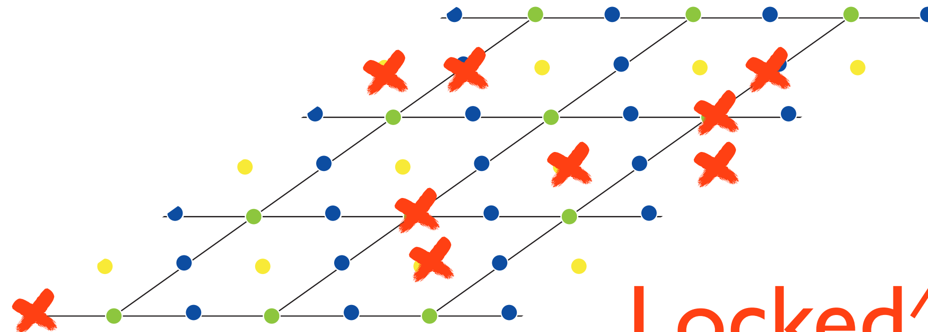
19 steps

SCHEDULING

Perfect lattice

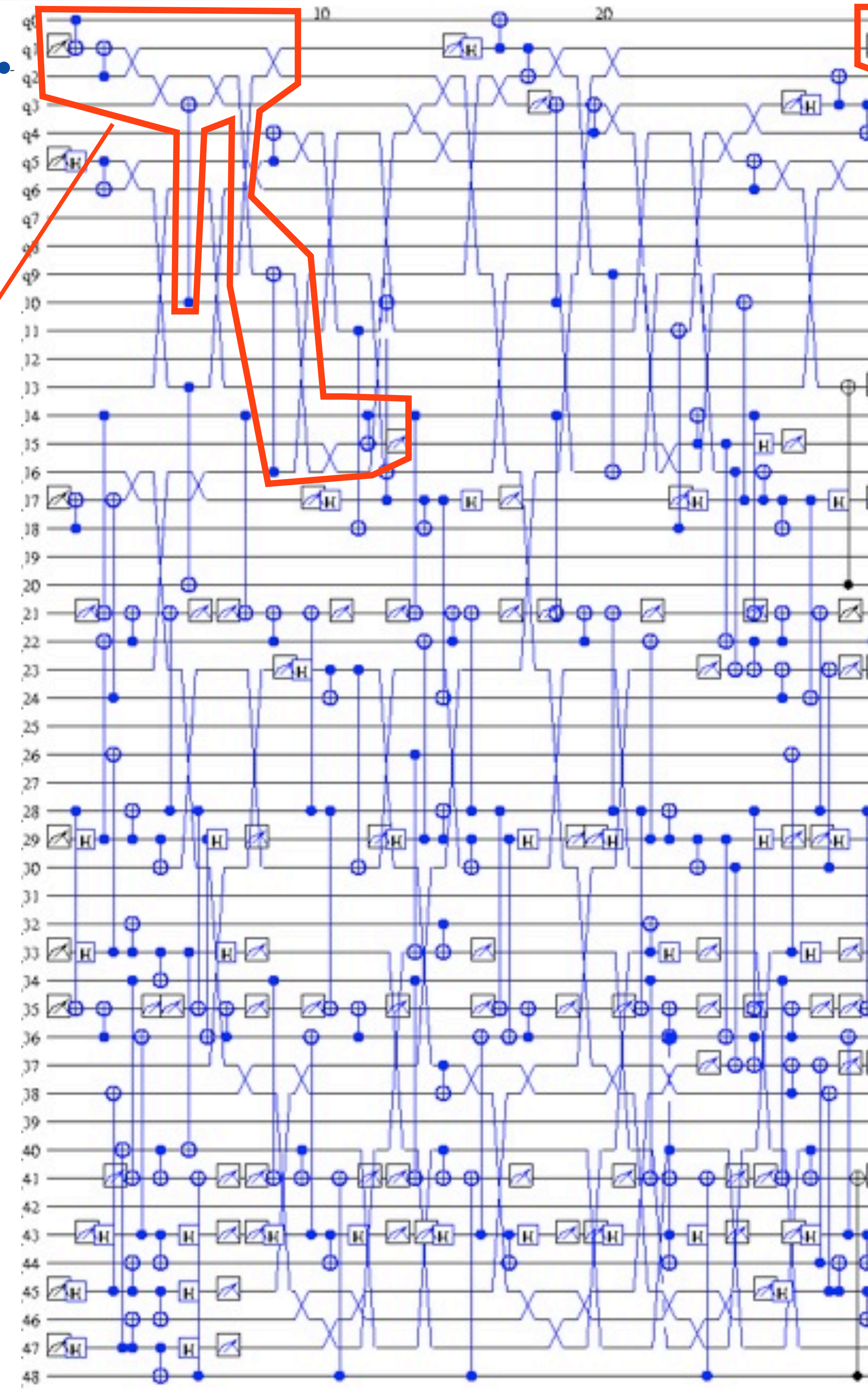


Defective lattice

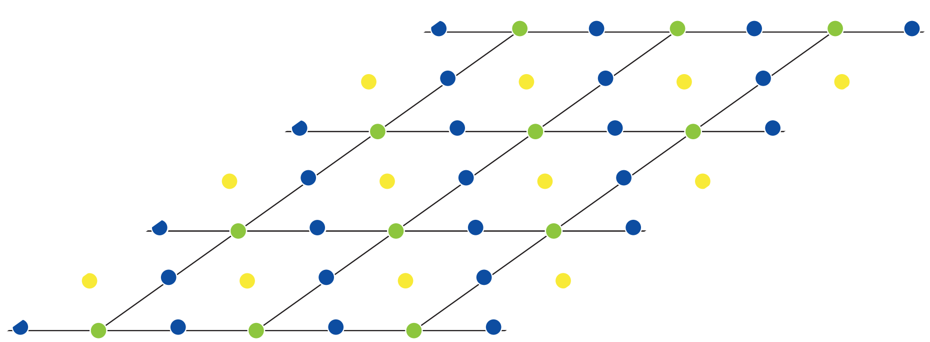


30steps/round

Locked



Remarkable point:
#step/round of error correction



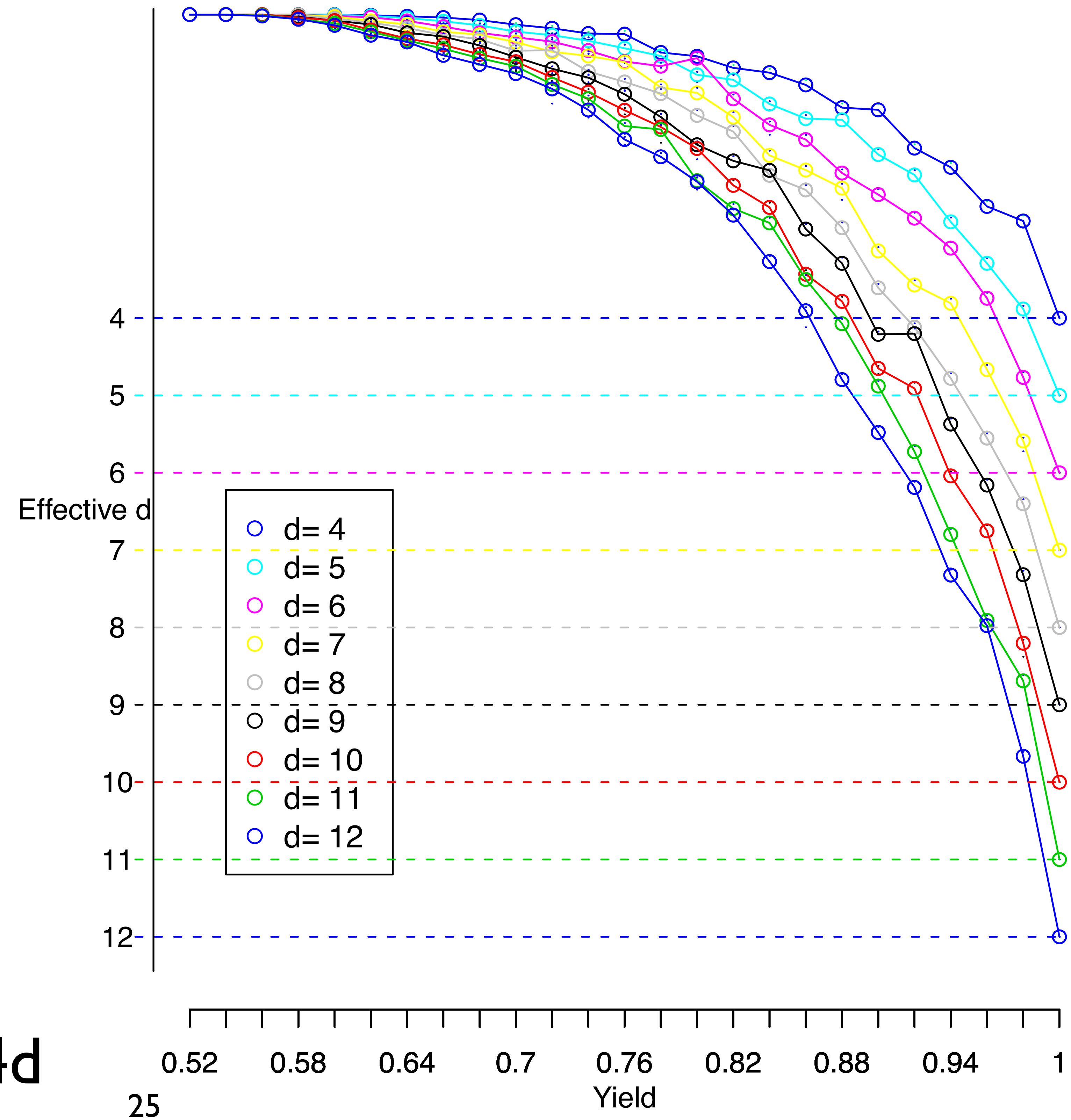
7steps/round

Defective lattice

Current estimate is that yield of 90% *halves* the effective code distance.

Nagayama *et al.*, in preparation

code distance = 4d



25

Plans for 2012

- Continue pushing surface code toward feasible experiments
- Resource estimates for Trotterized phase estimation (quantum simulation algorithm)
  - Condensed matter ground/excited state energies
  - Examine robustness against error

8 Hours of Lecture on Surface Code

- http://aqua.sfc.wide.ad.jp/Publications.html

Wikipedia QKD Article

量子鍵配送 - Wikipedia

ja.wikipedia.org/wiki/量子鍵配送

ウィキペディア
フリー百科事典

メインページ
コミュニティ・ポータル
最近の出来事
新しいページ
最近の更新
おまかせ表示
練習用ページ
アップロード (ウィキメディア・コモンズ)

ヘルプ
ヘルプ
井戸端
お知らせ
バグの報告
寄付
ウィキペディアに関するお問い合わせ

ツールボックス

他の言語
English

ページ ノート 閲覧 編集 履歴表示 検索

量子鍵配送

量子鍵配送(Quantum Key Distribution, QKD)は、通信を行う二者間でのセキュア通信を保証するために、量子力学を用いてラ
共有し、それをもとに情報を暗号・復号化する。量子鍵配送はしばしば量子暗号と混同されるが、量子鍵配送は量子暗号技術の
る。

量子配送を利用することによって得られる重要な性質は、通信を行う二者がその通信に用いられる鍵情報を取得しようとする盗
できるという点である。これは量子力学の基本的原理によるもので、量子系は観測することによってそれ自体が分散してしまう
する第三者は何らかの方法で鍵の情報を観測する必要があり、その観測行為が探知可能な片側性を引き起こすことを利用する。
もつれを用い、量子状態にある情報を転送することによって傍受を探知することが出来る通信システムを実装することが出来る
きい値を下回ったとき、秘匿性が保証された暗号鍵を生成し、それ以外の場合は傍受が行われたとして鍵生成は行わずに通信を

QKDにおける秘匿性は量子力学の原理を根拠にしているのに対し、従来の暗号鍵配送プロトコルは逆関数の計算が非常に困難で
根拠としているため、傍受を探知することが出来ず、またそれ故に秘匿性を完全に保証することは出来ない。

量子鍵配送は鍵を生成・配送することにのみ使われ、実際のデータ転送には使われない。すなわちこの暗号鍵はどんな暗号化ア
ることができ、暗号化されたデータは通常の伝送路によって送ることが出来る。

これに最も適した暗号化アルゴリズムとしてワンタイムパッドがあり、これは不規則な秘密鍵を用いた際に証明可能安全性を持
られている。[1]

目次 [非表示]

1 量子鍵交換
1.1 BB84 protocol: Charles H. Bennett and Gilles Brassard (1984)
1.2 E91 protocol: Artur Ekert (1991)