A no-go Ptheorem for fault-tolerant AQC

Robin Blume-Kohout,
Kevin Young, and Mohan Sarovar

Sandia National Laboratories

U.S. DEPARTMENT OF ENERGY
Sandia National Laboratories

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Based on *PRX* 3, 041013 (2013)
a.k.a. arXiv:1307.5893

What's a "Ptheorem"?

- A term I made up to convey the idea of a compelling but not-yet-rigorous argument.

- A physicist's theorem. In the spirit of:

$$\{=_p\} \; =_p \{=_m\} \neq_m \{=_p\}$$

- A pseudo-theorem.

What I mean by AQC

- [An] adiabatic quantum computation.

What I mean by AQC

- [An] adiabatic quantum computation.
    - Not holonomic (*fine*).
    - Not dissipative (*promising*).
    - Not [noise-driven] “annealing” (*see “dissipative”*).
    - Not GSQC (*has different problems*).
    - Not digital simulation of AQC on a FTQC (*fine*).
    - In this talk, AQC means:

      (1) computation driven by slowly varying Hamiltonian.

      (2) unique ground state <u>*or*</u> any ground state is a solution.

arXiv.org > quant-ph > arXiv:1406.2690

Quantum Physics

Adiabatic topological quantum computing

Chris Cesare, Andrew J. Landahl, Dave Bacon, Steven T. Flammia, Alice Neels

Fault Tolerance

- A uniform mapping of ideal computations to physical implementations with a specific noise model so that:

- For any problem size (N) and <u>certain</u> nonzero noise rates (ε), p_{fail} can be made arbitrarily small, and

- Only resources within the computational model are used (e.g., no fast gates for AQC), and

- The *amount* of resources used scales reasonably (usually polynomially) with N and $\log(p_{\text{fail}})$.

The core problem
(summary)

1. Typical AQCs (e.g. QUBO) are not robust against realistic noise unless they are *encoded* in a QECC.

2. The distance of the QECC must scale with N.

3. High-distance QEC negates low-weight operations.

4. AQC must be driven by slowly-varying low-weight Hamiltonians (Nature didn't provide high-weight).

5. So QEC negates the AQC-driving dynamics.

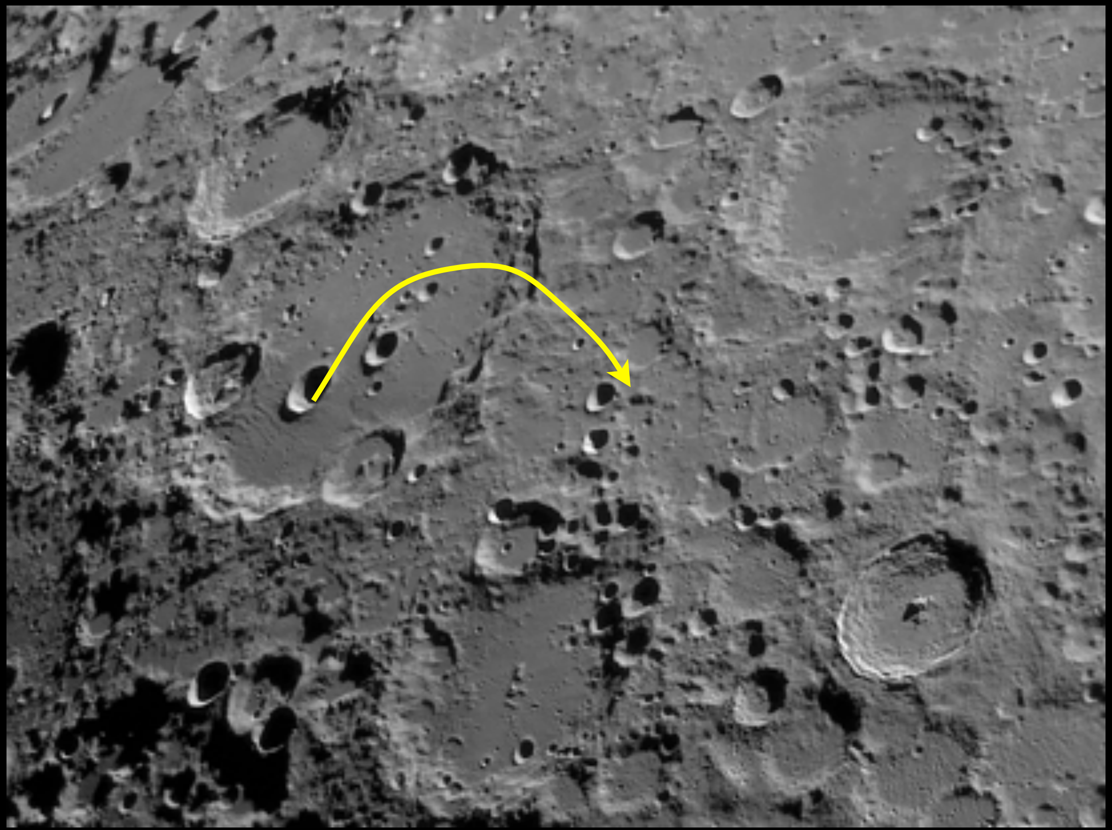
The core problem
(summary)

1. Typical AQCs (e.g. QUBO) are not robust against realistic noise unless they are *encoded* in a QECC.
2. The distance of the QECC must scale with N.
3. High-distance QEC negates low-weight operations.
4. AQC must be driven by slowly-varying low-weight Hamiltonians (Nature didn't provide high-weight).
5. So QEC negates the AQC-driving dynamics.
   (Except for a limited set of operations that are not sufficiently rich to drive logical (encoded) AQC -- see later...)

AQC is not naturally FT

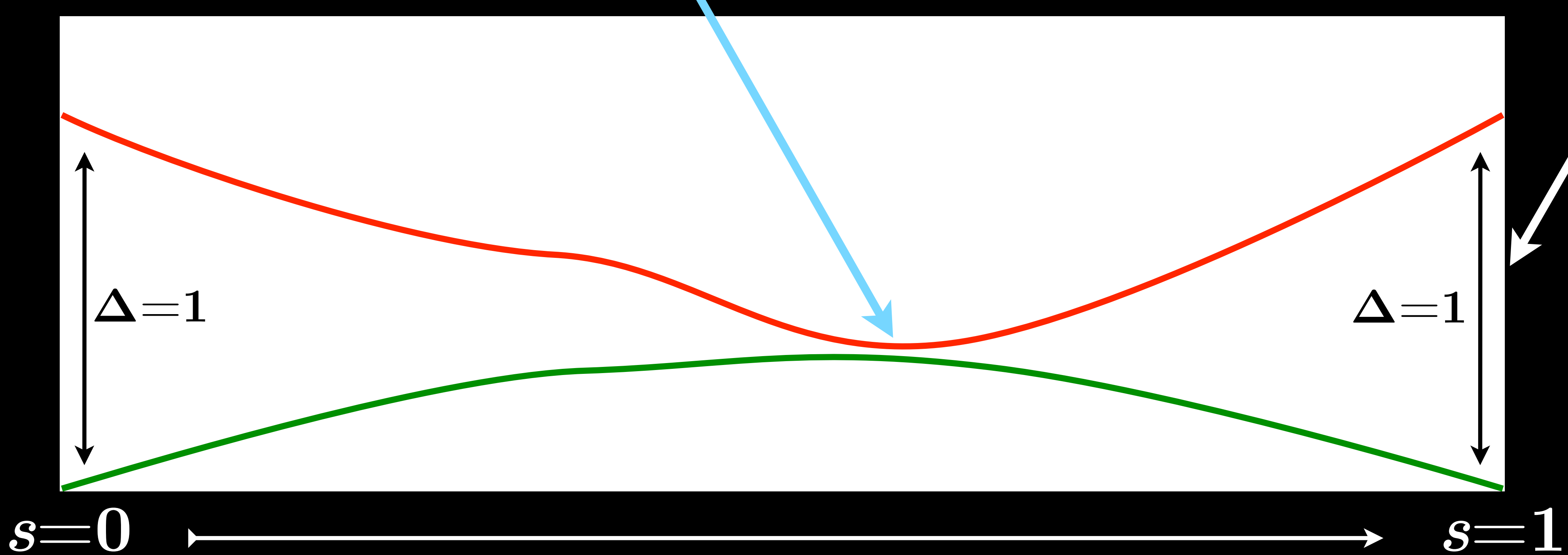
- AQC is robust to *several* kinds of error.
  - Dissipation, dephasing in eigenbasis of $\boldsymbol{H}$, smooth variations in control path.
- But there are other errors -- notably, i.i.d. local bit/phase flips -- to which AQC is *not* obviously robust.
- Gap protection is not enough -- gap is $\leq 1/\mathrm{poly}(N)$ at intermediate times (note: *local* gap matters).
- Local cooling isn't sufficient -- AQC Hamiltonians probably do not thermalize efficiently.

AQC is not naturally FT



phase transition (?) excitations at all length scales

doesn't thermalize fast



The Standard Noise Model

- AQC needs to be robust against (at least) uncorrelated local errors, e.g.:

  Each physical qubit has probability $p \ll 1$
  of experiencing a random Pauli (X, Y, Z).

  + coupling to a realistic cold bath (which actually helps)

- In a fixed timestep, we expect $pN = \Omega(N)$ errors.

- Hard optimization problems (3SAT) have “basins” of radius $o(N)$. So, cooling (local optimization) of unencoded AQC won’t recover the ground state.

- We must *encode* to make larger recoverable basins.

Encoding and QECC

- Encoding embeds N "logical" qubits into M physical qubits.
- The M-qubit physical Hilbert space factors:

$$\mathcal{H} = \mathcal{H}_{\text{logical}} \otimes \mathcal{H}_{\text{syndrome}}$$

- The 2^N-dimensional *code space* contains:

$$\left\{ |\psi\rangle_{\text{logical}} \otimes |0\rangle_{\text{syndrome}} \right\}$$

- Starting in the code space, <u>every</u> operation on at most d (the code *distance*) physical qubits acts <u>only</u> on the syndrome:

$$E\,|\psi_l 0_s\rangle = |\psi_l \phi_s\rangle$$

- We need d=O(N) to protect against O(N) errors per cycle.

How QEC works

- Let's be generous.
- Assume we have a Magic Black Box that just *does* perfect error correction every clock tick (e.g. 1 μs).
- So:

  (1) During a cycle, errors map $|\psi\rangle_l \, |0\rangle_s \rightarrow |\psi\rangle_l \, |\phi\rangle_s$

  (2) Then, QEC maps $|\psi\rangle_l \, |\phi\rangle_s \rightarrow |\psi\rangle_l \, |0\rangle_s$
- Now we just have to implement *logical* AQC.

How AQC works

- Start in the ground state of an X-type Hamiltonian

$$H_0 = \sum_k X^{(k)}$$

- Finish by applying a Z-type "problem" Hamiltonian

$$H_f = \sum_k a_k Z^{(k)} + \sum_{j,k} b_{jk} Z^{(j)} Z^{(k)}$$

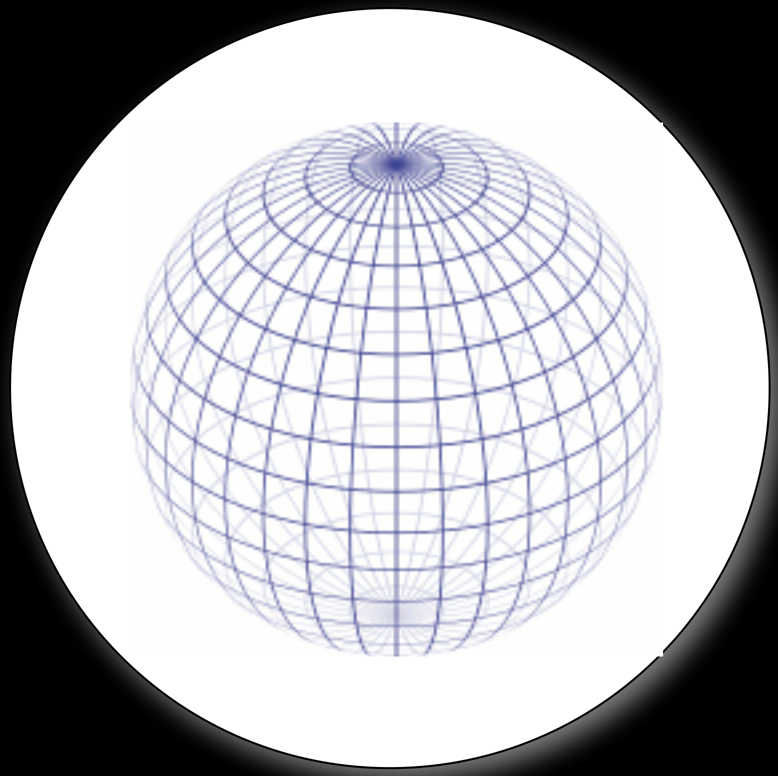
- Slowly change from H_0 to H_f. Drag the ground state along.

$$H(s) = (1-s)H_0 + sH_f$$

- But why does this work -- why does it stay in the ground state of the [changing] Hamiltonian?

How AQC works

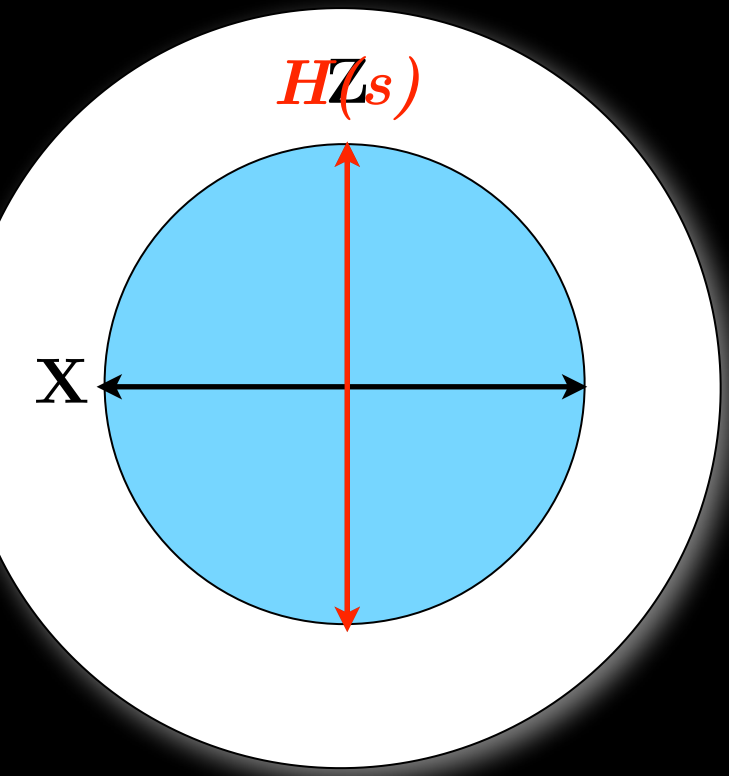
- <u>One</u> way to understand AQC is as via the Zeno effect:
  (1) Change $\boldsymbol{H}$ slightly.
  (2) Evolve for a random time.
  (3) GOTO 1.



- Random time evolution by $\boldsymbol{H}$ *dephases* in the [new] energy eigenbasis. For small change in $\boldsymbol{H}$, collapse into the [new] ground state is almost certain.

How AQC works

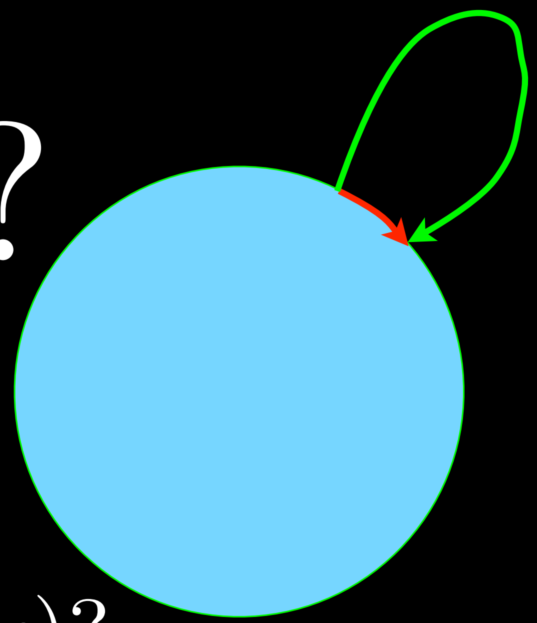
- To drag the ground state, we must dephase in a series of closely related bases.
- This means either *rotating around* or *measuring in* the basis of $\boldsymbol{H}(s)$ at closely spaced times s_1, s_2,...
- When $\boldsymbol{H}(s)$ is implemented directly on physical qubits, this is easy.
- Doing it *logically* is a problem.



Logical AQC

- The QEC defines $\mathcal{H} = \mathcal{H}_{\text{logical}} \otimes \mathcal{H}_{\text{syndrome}}$.
- Any operator with *weight* less than $d = \mathrm{O}(N)$ will act only on the syndrome.
  (weight = # of physical qubits on which it acts nontrivially)
- We must map the original AQC Hamiltonian to an equivalent *logical* Hamiltonian that acts on $\mathcal{H}_{\text{logical}}$.
- **Such a logical H must have weight $\mathrm{O}(N)$.**
  (Terms of weight $< d$=$\mathrm{O}(N)$ will be negated by QEC!)
- Sadly, available Hamiltonians have weight $\mathrm{O}(1)$. ☹

Can we simulate H?

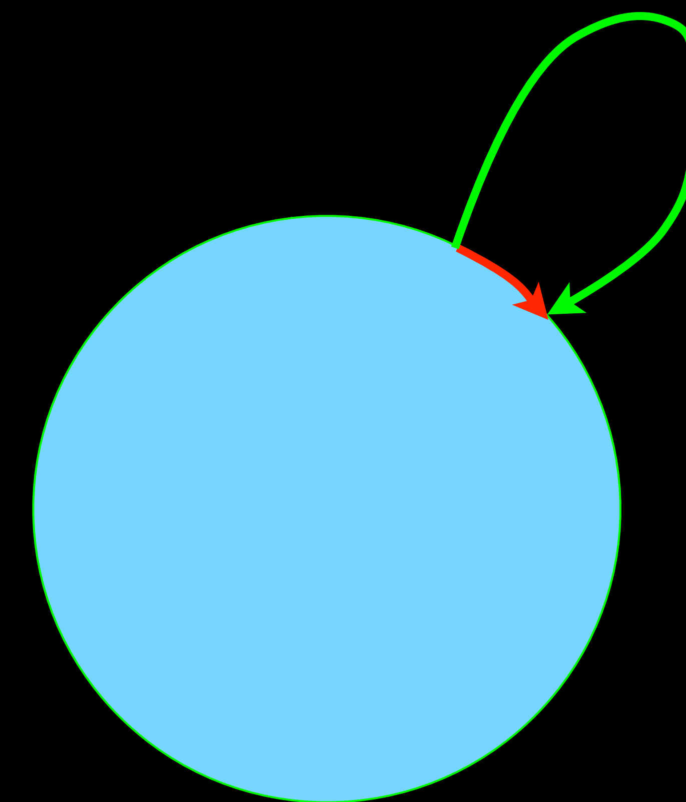


- Maybe we don't need to map $H(s)$ to $H_{\text{logical}}(s)$?
- We just need to *simulate* $H(s)$ on the code space.
- A nice example:

  Suppose our code has a transversal X gate (like the Ising or toric codes): $X_{\text{logical}} = \bigotimes_k X_k$

  Now, $H(s) = X_{\text{logical}}$ generates a logical bit flip. But so does $H = \sum_k X_k$!
- We can leave the code space *between* QEC cycles.

Can we simulate H?



- We can leave the code space between QEC cycles. Can we take advantage of this to simulate the AQC $H(s)$? **No**.

- We need to dephase in the eigenbasis of $H(s)$ at s_0, $s_0+\epsilon$, $s_0+2\epsilon\ldots$

- Suppose $e^{iH(s_0)\delta t}$ is a logical operation. Then

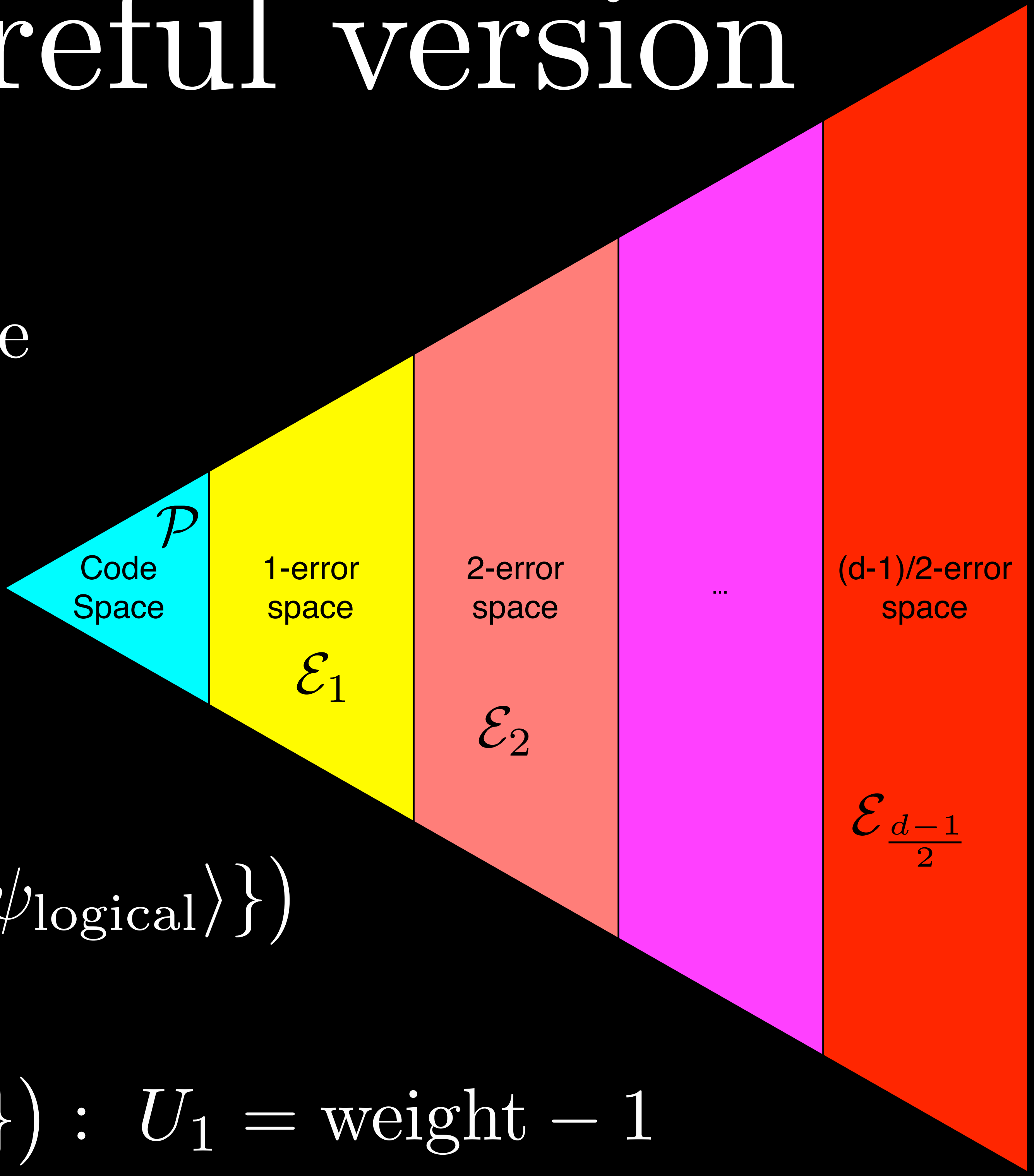
$$e^{iH(s_0+\epsilon)\delta t} = e^{i[H(s_0)+\epsilon\Delta H]\delta t} \approx \underbrace{\boxed{e^{iH(s_0)\delta t}}}_{\text{Logical}}\underbrace{\boxed{e^{i\epsilon\Delta H\,\delta t}}}_{\text{Error}}$$

This isn't actually technically correct.

- Small changes in $H_{\text{physical}} \neq$ small changes in H_{logical}.

A more careful version

This is what the Hilbert space of an error correcting code looks like:



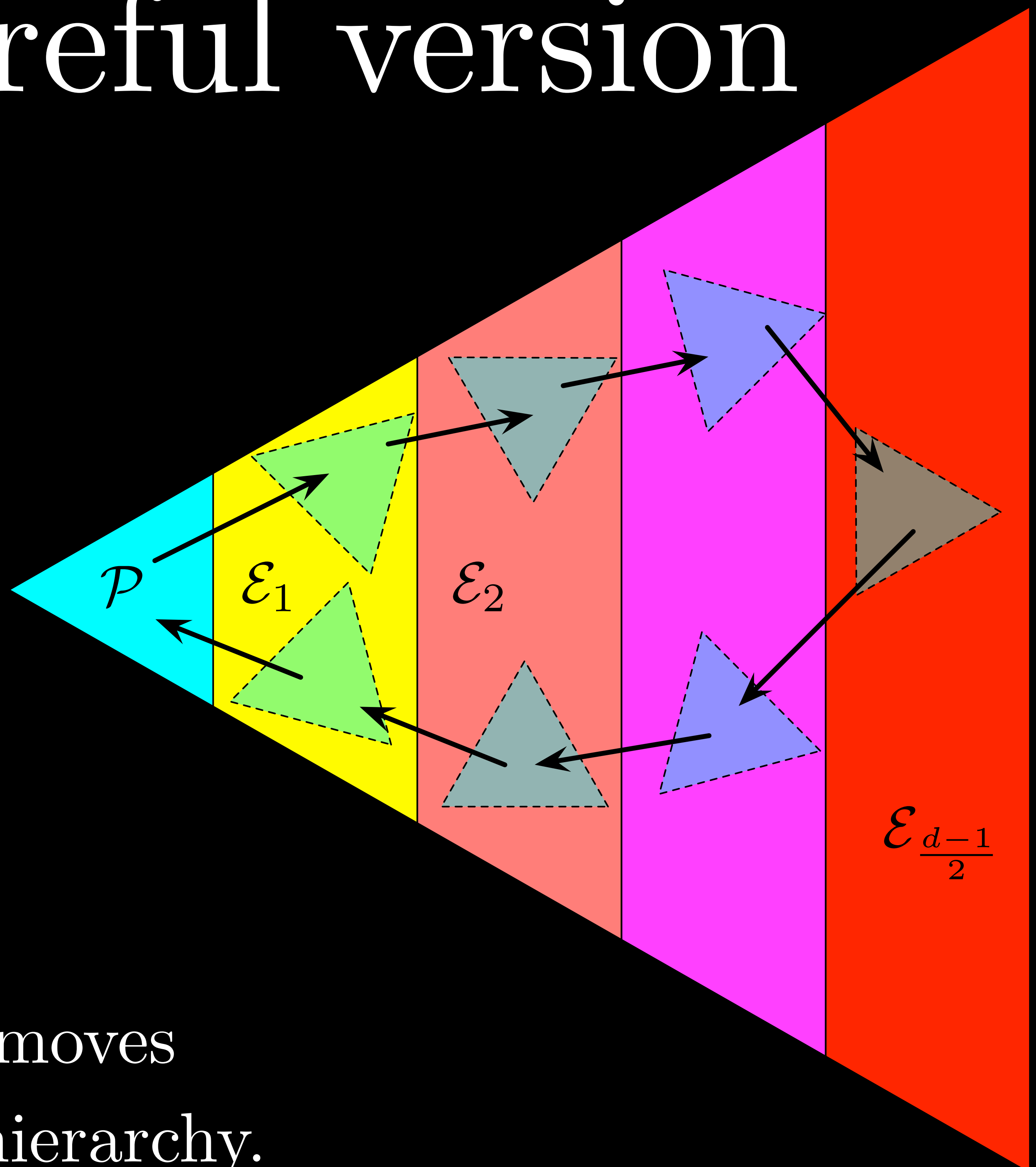
$$\mathcal{P} = \mathrm{Span}\big(\{|\psi_{\mathrm{logical}}\rangle\}\big)$$

$$\mathcal{E}_1 = \mathrm{Span}\big(\{U_1\,|\psi_{\mathrm{logical}}\rangle\}\big) :\ U_1 = \mathrm{weight} - 1$$

$$\mathcal{E}_w = \mathrm{Span}\big(\{U_w\,|\psi_{\mathrm{logical}}\rangle\}\big) :\ U_w = \mathrm{weight} - \mathrm{w}$$

A more careful version

This is what a transversal implementation of a logical unitary looks like:

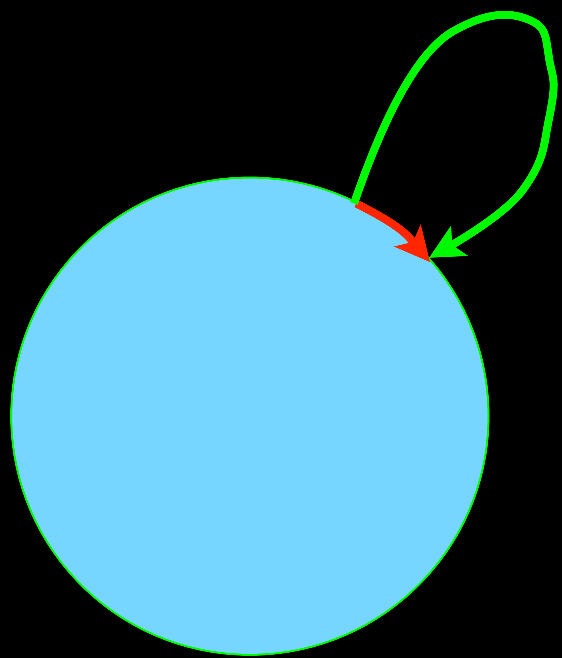


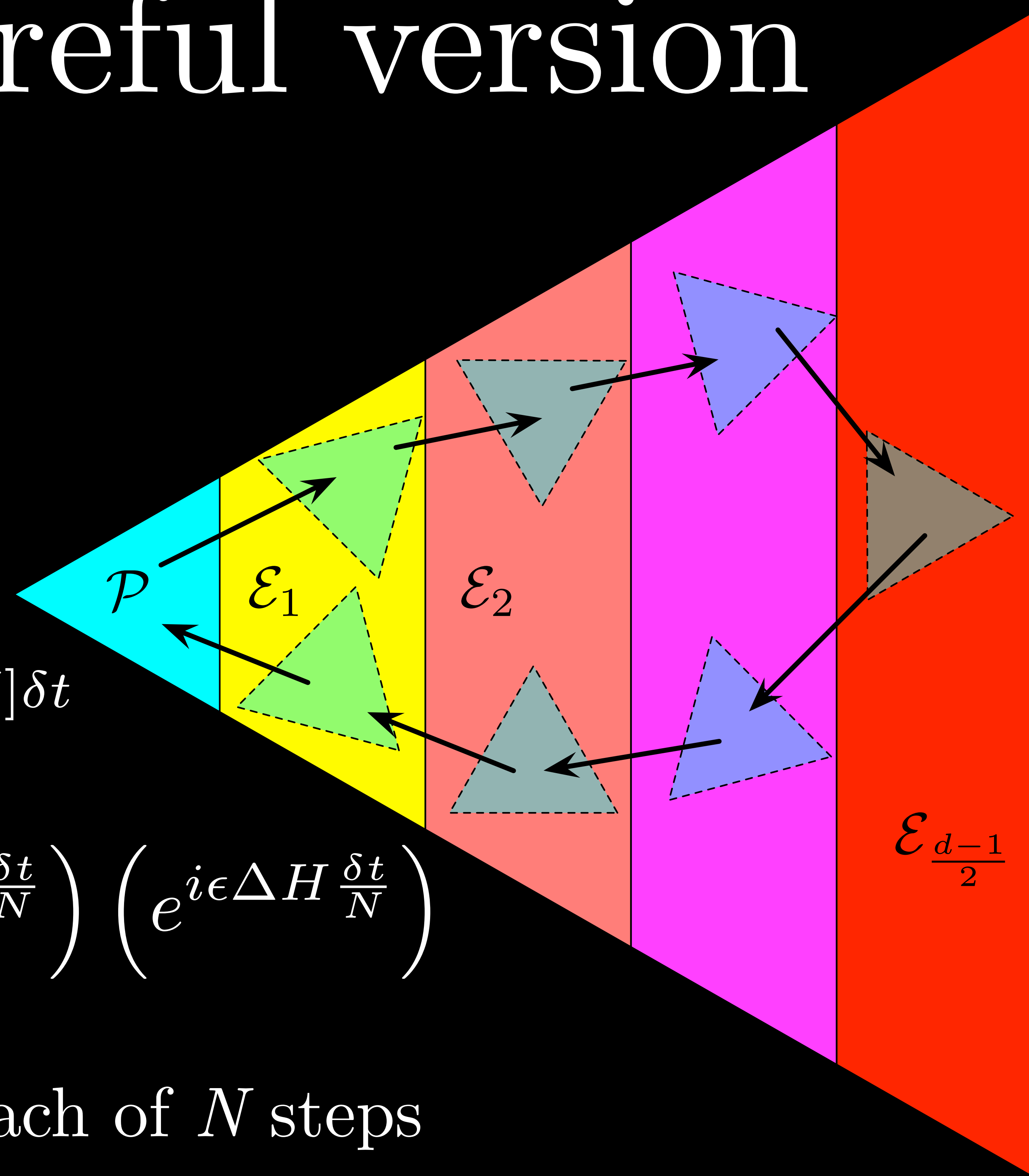
Our code distance d is O(N).

Each weight-w operation only moves us w steps on the error-space hierarchy.

Logical operations require “loops” to w=(d-1)/2 and back.

A more careful version





$$e^{iH(s_0+\epsilon)\delta t} = e^{i[H(s_0)+\epsilon\Delta H]\delta t}$$

$$\approx \prod_{k=1}^{N} \left(e^{iH(s_0)\frac{\delta t}{N}}\right)\left(e^{i\epsilon\Delta H\frac{\delta t}{N}}\right)$$

1. Tiny amplitude for ΔH at each of N steps

-- can't add up to a full "loop" on its own.

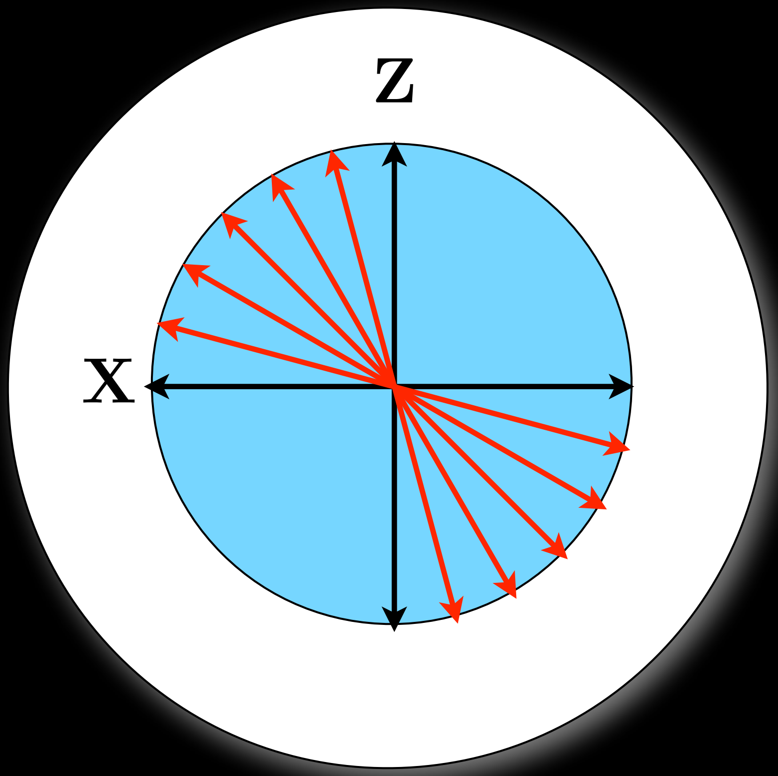
2. If a small, low-weight term at *any* intermediate step can cause a logical operation, then the code is not doing its job!

A stronger argument

- There is an even stronger no-go argument for "simulating" logical AQC encoded in *topological stabilizer* codes using *spatially local* Hamiltonians.

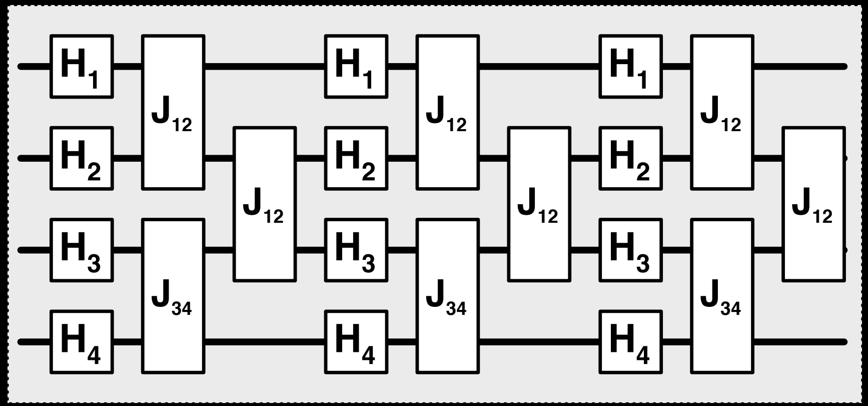
- Suppose we give up on smoothly mapping $\boldsymbol{H}(s)$ --> $\boldsymbol{H}_{\text{logical}}(s)$. Let $\boldsymbol{H}(s)$ be totally different at each timestep.

- Now, can we <u>simulate</u> $\boldsymbol{H}_{\text{AQC}}(s)$ that is:
  (1) non-commuting (for $s \neq s'$)
  (2) continuously varying w/r.t. s?



A different argument

- Action of local & low-weight Hamiltonian for constant time is well approximated by a *constant depth circuit*: (just Trotterize it...)



- But Bravyi and Koenig (*PRL* 110, 170503; 2012) proved a generalization of Eastin-Knill: In any *topological stabilizer code*, the unitary operations that can be implemented by constant-depth circuits form a discrete set (Cliffords, in D=2 spatial dimensions).

- Simulating AQC is incompatible with topological QEC.

*Reed-Muller codes allow a *commuting* almost-continuous set

Summary

- Typical AQCs (e.g. QUBO) are not robust against realistic noise unless they are *encoded* in a QECC.
- The distance of the QECC must scale with N.
- High-distance QEC negates low-weight operations.
- AQC must be driven by slowly-varying low-weight Hamiltonians (Nature didn't provide high-weight).
- So QEC negates the AQC-driving dynamics.
  (Except for a limited set of operations that are not sufficiently rich to drive logical AQC.)

Based on *PRX* 3, 041013 (2013)
a.k.a. arXiv:1307.5893

Loopholes?

- There *are* various potential loopholes that we haven't closed yet. Some involve redefining AQC:
  - Build a fault-tolerant circuit-model QC, and use it to simulate an AQC (by Trotterization). *But why?*
  - Dissipative computing -- where noise helps drive computation -- seems promising. *Just don't call it AQC*.
  - Adiabatic *gates* are fine (holonomic). *Really not AQC*.
  - GSQC? *Neat idea, but Matt's objection is compelling...*

Technical Loopholes?

- Perhaps exotic codes -- not topological, or not stabilizer -- could break Bravyi-Koenig? *Seems unlikely -- Eastin-Knill applies to all codes, and has a similar flavor. Would be revolutionary.*

- Perhaps we could design specialized codes with "holes" through which low-weight Hamiltonians could drive computation? *But then errors would get through too...*

- New perturbative gadgets that simulate weight-N Hamiltonians at reasonable (subexponential) cost? *Even if this were possible, gadgets are generally not fault-tolerant, because they use unprotected ancillae...*

- Tricks like Paetznick/Reichardt or Jochym-O'Connor/Laflamme? *Probably not - these give <u>universal</u> gatesets, not continuous ones.*