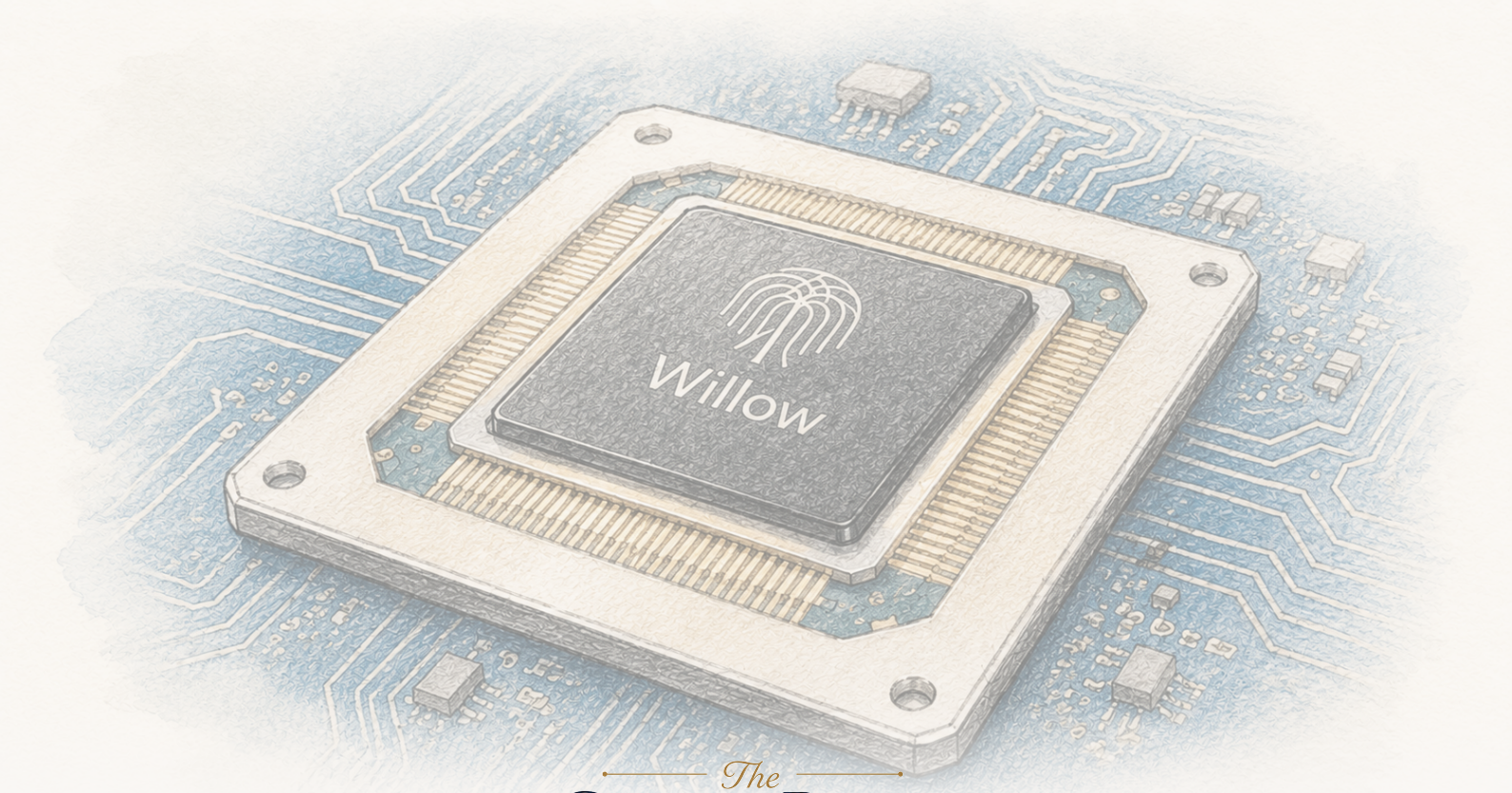


The Clean Paper

WHAT THE PAPER SAYS · WHAT IT DOESN'T · WHY IT MATTERS

Квантова пам'ять нарешті стала кращою зі збільшенням — справжня віха, але ще не готова машина

Google Quantum AI вперше чисто показала, що квантова пам'ять на surface code може працювати нижче порога: зі збільшенням коду від distance 3 до 5 і 7 логічна частота помилок експоненційно падала — приблизно у 2,14 раза на кожні два кроки, — а найбільша distance-7 пам'ять на 101 кубіті переживала свій найкращий фізичний кубіт, тобто вийшла beyond breakeven, при корекції помилок у реальному часі. Це давно очікувана інженерна віха. Але це також лише один логічний кубіт у ролі пам'яті, з частотою помилок усе ще далеко від потреб реальних алгоритмів, без логічних операцій, із незрозумілою підлогою помилок, яку підкреслюють автори, і з багатьма порядками масштабування попереду. Поріг перетнуто, а не готовий комп'ютер доставлено.

Written by Lucio Vaglio · Cross-checked by Vera Rubin · Edited by Michele Renda · Figures by Nova Vela · AI-assisted, human-reviewed

Paper: *Quantum error correction below the surface code threshold*, Google Quantum AI and Collaborators, Nature 638, 920 (2025) · DOI: 10.1038/s41586-024-08449-y

Момент, коли крива нарешті зігнулася в правильний бік

Тридцять років квантова корекція помилок трималася на обіцянці, яку ще ніколи не виконували настільки чисто. Теорія каже: якщо розподілити одну одиницю квантової інформації — один *логічний* кубіт — між багатьма шумними фізичними кубітами, і якщо ці фізичні кубіти достатньо якісні, то додавання *ще більшої* їх кількості має робити логічний кубіт *кращим*, а помилки повинні спадати експоненційно зі зростанням коду. Умова захована в «якщо»: нижче критичного порога шуму більша кількість кубітів допомагає; вище нього вона лише додає шуму. Усі попередні експерименти жили не по той бік цієї межі або не показували тренд достатньо чисто. Збільшення коду робило ситуацію гіршою, а не кращою.

У грудні 2024 року Google Quantum AI повідомила про першу чітку демонстрацію іншого режиму. На Willow, новому поколінні надпровідних процесорів компанії, дослідники побудували пам'ять із surface code на відстанях коду 3, 5 і 7 й побачили, як логічна частота помилок *падає* щоразу, коли код стає більшим — у $\Lambda = 2{,}14 \pm 0{,}02$ раза на кожне збільшення відстані на два. Найбільша система — пам'ять distance-7 на 101 кубіті — утримувала логічний кубіт із помилкою $0{,}143\% \pm 0{,}003\%$ за цикл корекції й, що стало окремим заголовком усередині заголовка, жила *довше за найкращий фізичний кубіт*, з якого була побудована, у $2{,}4 \pm 0{,}3$ раза. Це називають «beyond breakeven» — виходом за точку беззбитковості, — і це перший випадок, коли весь апарат корекції помилок на цьому обладнанні справді окупив власні накладні витрати.

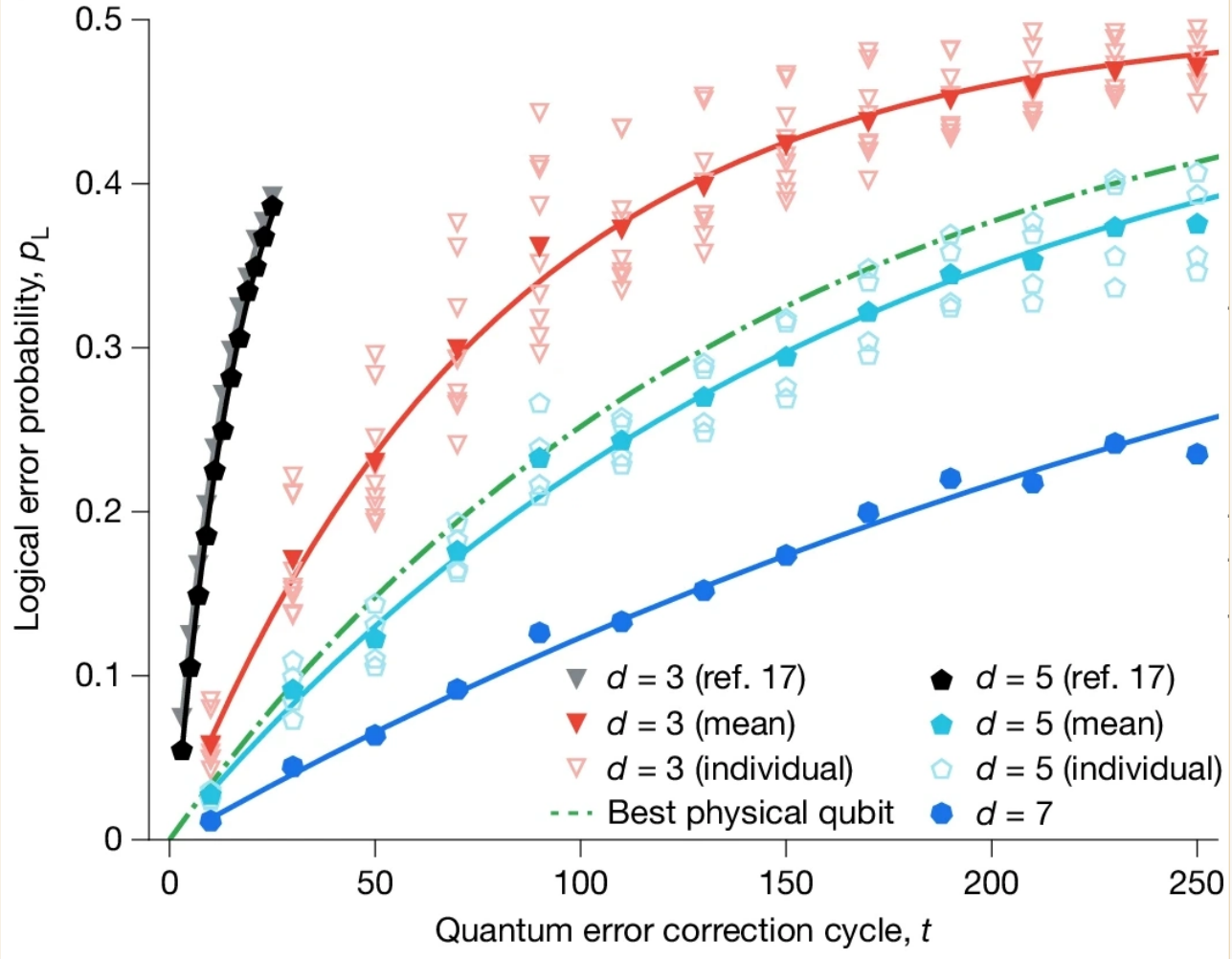
Це справжня віха, і важливо точно сказати, якого типу. Це доказ того, що масштабування тепер іде в правильний бік. Це не готовий квантовий комп'ютер, і стаття цього не стверджує.

Що означають «surface code», «distance» і «below threshold»

Логічний кубіт — одна захищена одиниця квантової інформації, закодована в багатьох фізичних кубітах. **Surface code** — конкретний спосіб такого кодування на двовимірній решітці, де додаткові «вимірювальні» кубіти постійно перевіряють наявність помилок, не руйнуючи збережену інформацію. **Відстань коду** d — не фізична відстань: це найменша кількість правильно розташованих помилок, здатних зіпсувати логічний кубіт так, щоб код цього не помітив. Більше d означає більшу й стійкішу ділянку коду: вона витрачає більше фізичних кубітів — приблизно $2d^2 - 1$ — і виправляє більше одночасних помилок, до $(d - 1)/2$. Отже, три перевірені тут розміри — відстані 3, 5 і 7 — виправляють відповідно 1, 2 і 3 одночасні помилки й теоретично потребують приблизно 17, 49 та 97 фізичних кубітів; побудована Google пам'ять distance-7 використовувала 101 — трохи більше за цей підручниковий мінімум.

«**Below threshold**» — ключова фраза. Корекція помилок допомагає лише тоді, коли фізична частота помилок лежить нижче критичного значення; у цьому режимі кожне збільшення відстані експоненційно пригнічує логічну частоту

помилок. Коефіцієнт пригнічення Λ вимірює саме це: $\Lambda > 1$ означає, що збільшувати код корисно, а що більше Λ, то краще. Google повідомляє $\Lambda \approx 2{,}14$, тобто кожне збільшення відстані на два приблизно вдвічі зменшувало логічну частоту помилок. Сам факт, що Λ впевнено вище 1, — і є центральним результатом.



Як логічна помилка накопичується протягом циклів корекції для пам'яті з відстанями 3, 5 і 7 — згори вниз. Лінія, на яку варто дивитися, — зелена штрихова: *найкращий одиночний фізичний кубіт* на чипі. Пам'ять distance-7 — синя, найнижча — накопичує помилку повільніше за цю лінію, тож закодований кубіт живе довше, ніж найкращий фізичний кубіт, з якого він побудований: «beyond breakeven», у 2,4× раза. Це результат щодо часу життя; саме пригнічення нижче порога ($\Lambda = 2{,}14$ при зростанні коду від distance 3 до 5 і 7) міститься в числах у тексті.

Google Quantum AI and Collaborators / Nature CC BY-NC-ND 4.0

Що зробили автори

- Побудували surface-code пам'ять на двох чипах Willow: 105-кубітному процесорі, який запускав коди distance-3, -5 та -7 для тесту масштабування — найбільша конфігурація була 101-кубітною distance-7 пам'яттю з 49 кубітами даних, — і 72-кубітному процесорі, який запускав distance-5 пам'ять із декодером реального часу та repetition codes великої відстані.
- Виміряли, як логічна помилка *на цикл* змінюється зі збільшенням відстані коду від 3 до 5 і 7, та отримали коефіцієнт пригнічення Λ.
- Порівняли час життя логічного кубіта з найкращим окремим фізичним кубітом на тому самому чипі, щоб перевірити «breakeven».
- Запустили distance-5 код із **декодером реального часу** — класичним обладнанням, яке інтерпретує перевірки помилок у міру їх появи, — до мільйона циклів, щоб показати, що корекція здатна працювати в темпі самої машини.
- Розширили простіші **repetition codes** до distance 29, щоб шукати рідкісні глибинні джерела помилок, які задають нижню межу продуктивності.

Що вони виявили

- **Код працює нижче порога.** Логічна помилка за цикл зменшувалася в $\Lambda = 2{,}14 \pm 0{,}02$ раза на кожне збільшення відстані на два — чітке експоненційне пригнічення, саме та поведінка, яку обіцяла теорія й яку жоден процесор раніше не демонстрував настільки однозначно.
- **Пам'ять distance-7 досягла 0,143% ± 0,003% помилки за цикл** і жила **у 2,4 ± 0,3 раза довше**, ніж її найкращий фізичний кубіт — beyond breakeven.
- **Декодування в реальному часі встигало.** Для distance 5 середня затримка декодера становила 63 мікросекунди за тривалості циклу 1,1 мікросекунди й стабільно працювала протягом мільйона циклів — корекція виконувалася наживо, а не лише в аналізі після експерименту.
- **Залишається рідкісне глибинне джерело помилок.** У тестах repetition code продуктивність зрештою обмежували корельовані сплески помилок приблизно **раз на годину** — близько одного випадку на 3×10^9 циклів, — що створює підлогу помилок поблизу 10^{-10}; походження цих сплесків, за словами авторів, **поки не зрозуміле**.

Чого це не доводить

- Це **не** квантовий комп'ютер, який виконує обчислення. Це квантова *пам'ять*: вона зберігає й захищає один логічний кубіт. Вона не виконує логічних операцій — gates — між логічними кубітами й не запускає алгоритмів.

- Це **не** означає, що до корисних машин залишився один кубіт. Логічний кубіт distance-7 витрачає близько 101 фізичного кубіта; частота помилок близько 0,1% за цикл усе ще набагато вища за приблизно 10^{-6}–10^{-10}, потрібні реальним алгоритмам. Закриття цього розриву вимагає набагато більших відстаней — значно більше фізичних кубітів на один логічний, — а корисним алгоритмам потрібні *тисячі* логічних кубітів одночасно. Фізичний бюджет для такого масштабу йде в мільйони.
- Фраза **«якщо масштабувати» справді несе вагу.** Власний висновок статті: продуктивність пристрою, *якщо її масштабувати*, може відповідати вимогам великих алгоритмів. Показати правильний тренд на одному логічному кубіті — не те саме, що побудувати масштабовану машину, і ніщо тут не гарантує, що тренд збережеться на значно більших розмірах.
- **Незрозуміла підлога помилок — жива проблема.** Корельовані сплески, які обмежують repetition code, за словами авторів, на порядки більші від очікуваного й завадили б більшим fault-tolerant застосуванням, доки їх не зрозуміють. Це відкрита вада, прямо визнана в роботі, а не дрібниця, яку вже розв'язано.
- Стаття нічого не говорить про **злам шифрування чи «квантову перевагу» в корисних задачах**. Для цього потрібна повна відмовостійка машина, фундаментом для якої є цей результат, а не сам цей демонстратор.

Наскільки сильні докази

- **Центральне твердження сильне й важливе.** Робота нижче порога із чітким експоненційним пригніченням на трьох відстанях коду, плюс beyond-breakeven час життя й робочий декодер у реальному часі — саме та комбінація, якої галузь намагалася досягти, і її показано безпосередньо, а не виведено непрямо. Це не артефакт хайпу, а реальний інженерний результат провідної групи.
- **Автори обережні щодо масштабу висновку.** Вони називають результат below-threshold *memory*, самі підкреслюють незрозумілу підлогу корельованих помилок і прив'язують майбутнє до помітного «if scaled». Перебільшення з'являється переважно в навколишньому медійному переказі, який округлює «захищений помилками кубіт пам'яті покращився зі збільшенням коду» до «квантові комп'ютери вже тут».
- **Чесний статус — фундаментальний крок, виконаний переконливо.** Один логічний кубіт, захищений настільки добре, що додаткова надмірність нарешті допомагає, — але попереду довгий, важкий і не гарантований шлях масштабування, логічних операцій та пояснення невідомих помилок.

Чому це важливо

Відмовостійкі квантові обчислення завжди мали присмак проблеми курки й яйця: корисним машинам потрібні частоти помилок, яких не здатен досягти жоден фізичний кубіт, а виправлення — корекція помилок — працює лише тоді, коли саме фізичне

обладнання вже достатньо добре, щоб опинитися нижче порога. Перетнути цю межу хоча б один раз, хоча б для одного логічного кубіта, змінює запитання з «чи це взагалі можливо?» на «наскільки далеко це можна масштабувати і як швидко?». Це реальна й змістовна зміна, тому результат заслуговує уваги.

Але та сама обережність, яка робить результат надійним, має стримувати історію навколо нього. Це перша добре покладена цеглина фундаменту. Не будівля — і самі люди, які її поклали, першими це визнають. Правильний спосіб стежити за квантовими обчисленнями найближчими роками — дивитися саме на цю неефектну криву: чи зберігається коефіцієнт пригнічення зі зростанням кодів, чи вдається виконувати логічні gates так само чисто, як логічну пам'ять, і чи буде колись пояснено ту загадкову помилку раз на годину.

Короткий підсумок

Google Quantum AI вперше чітко продемонструвала, що квантова пам'ять на surface code може працювати *нижче порога*: коли код збільшували від distance 3 до 5 і 7, логічна частота помилок експоненційно падала — приблизно у 2,14 раза на кожні два кроки, — а найбільша distance-7 пам'ять на 101 кубіті переживала свій найкращий фізичний кубіт, тобто вийшла beyond breakeven, при цьому корекція помилок працювала в реальному часі. Це справжня, давно очікувана інженерна віха для квантових комп'ютерів. Але це також один логічний кубіт у ролі пам'яті, з частотою помилок усе ще далеко від вимог реальних алгоритмів, без виконаних логічних операцій, із незрозумілою підлогою помилок, яку самі автори підкреслюють, і з багатьма порядками масштабування попереду. Справжній поріг перетнуто — квантовий комп'ютер ще не доставлено.

Джерела

Google Quantum AI and Collaborators, Quantum error correction below the surface code threshold, Nature 638, 920 (2025)
https://doi.org/10.1038/s41586-024-08449-y

Author Correction: Quantum error correction below the surface code threshold, Nature 653, E5 (2026) — corrects a Fig. 3a axis label and legend keys; does not affect the below-threshold (Fig. 1) results
https://www.nature.com/articles/s41586-026-10559-8