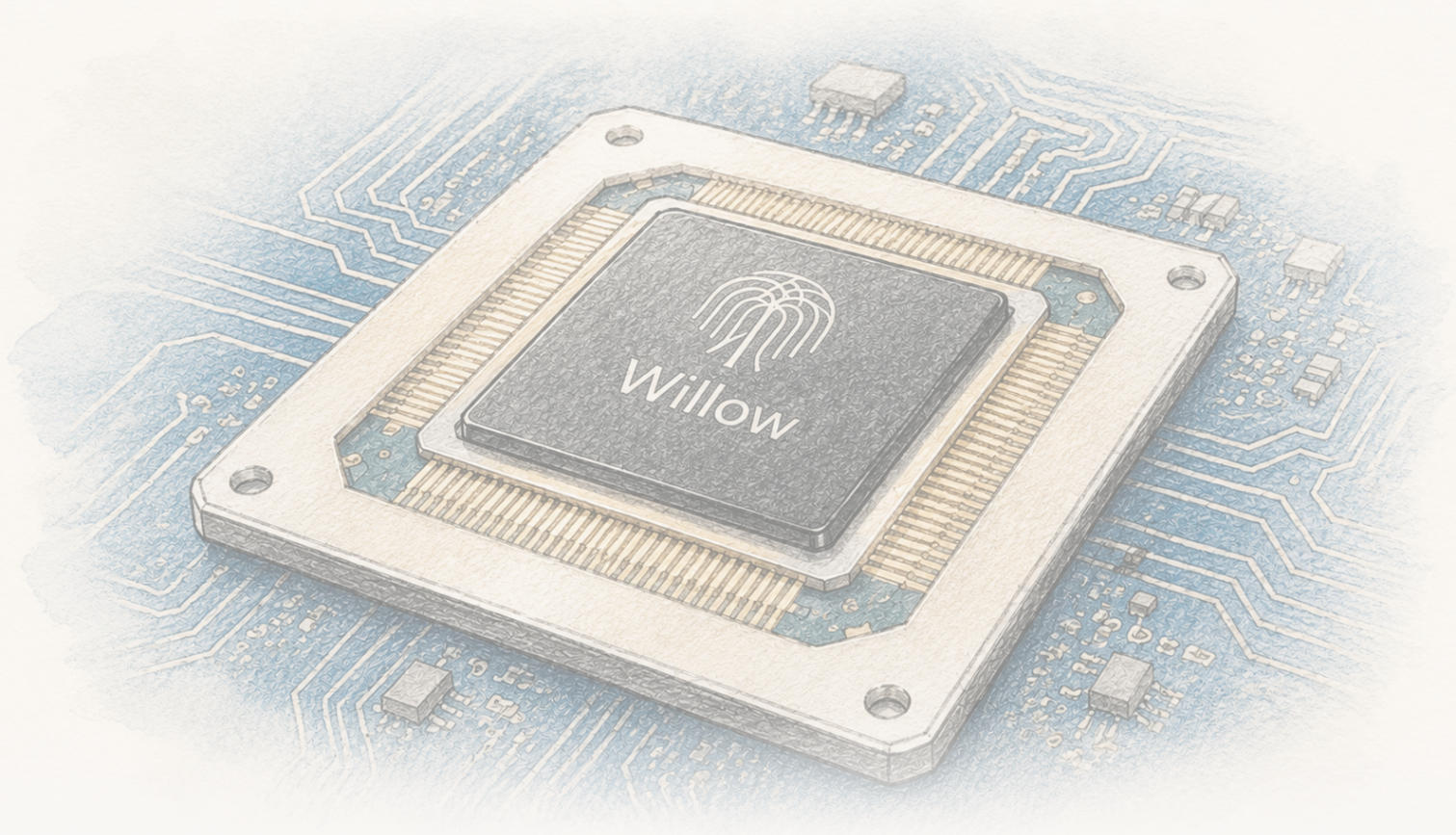




The CLEAN PAPER

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

ในที่สุด quantum memory ก็ดีขึ้นเมื่อขยายใหญ่ — หมุดหมายจริง และเครื่องที่มันยังไม่ใช่

Google Quantum AI แสดงอย่างสะอาดเป็นครั้งแรกว่า quantum memory แบบ surface code ทำงาน below threshold ได้: เมื่อขยาย code จาก distance 3 เป็น 5 เป็น 7 logical error rate ลดลงแบบเอ็กซ์โพเนนเชียล (ราว 2.14 เท่าต่อสองขั้น) และหน่วยความจำใหญ่สุดแบบ distance-7 ขนาด 101 qubits อยู่ได้นานกว่า physical qubit ที่ดีที่สุด — beyond breakeven — พร้อม error correction แบบ real time นี่สื่อหมุดหมายวิศวกรรมที่ตามหามานาน แต่ยัง เป็น logical qubit เพียงหนึ่งตัวในฐานหน่วยความจำ มี error rate ห่างจากที่อัลกอริทึมจริงต้องการมาก ไม่ได้ทำ logical operations มี error floor ที่ยังอธิบายไม่ได้ และยังต้อง scaling อีกหลายลำดับขนาด ข้าม threshold แล้ว ไม่ใช่ส่งมอบคอมพิวเตอร์แล้ว

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

ช่วงเวลาที่เส้นโค้งหันไปในทิศทางที่ต้องการ

ตลอดสามสัปดาห์ การแก้ไขข้อผิดพลาดควอนตัม ตั้งอยู่บนคำสัญญาที่ไม่เคยถูกทำให้เห็นอย่างสะอาด ทฤษฎีบอกว่า หากกระจายข้อมูลควอนตัมหนึ่งหน่วย — *logical* คิวบิต หนึ่งตัว — ไปบน คิวบิตกายภาพ ที่มี noise จำนวนมาก และหาก คิวบิตกายภาพ เหล่านั้นดีพอ การเพิ่ม คิวบิต *มากขึ้น* ควรทำให้ คิวบิตเชิงตรรกะ *ดีขึ้น* โดย ข้อผิดพลาด ลดลงแบบเอ็กซ์โพเนนเชียลเมื่อ code ใหญ่ขึ้น จุดติดอยู่ที่คำว่า “หาก”: ถ้าว่า noise threshold วิกฤต การเพิ่ม คิวบิต ช่วย; สูงกว่านั้น การเพิ่ม คิวบิต เพียงเพิ่ม noise การทดลองก่อนหน้าทั้งหมดอยู่ฝั่งด้านของเส้นนี้ หรือไม่ก็แสดงแนวโน้มได้ไม่ชัด เมื่อขยาย code สิ่งต่าง ๆ แยก ไม่ใช้ดีขึ้น

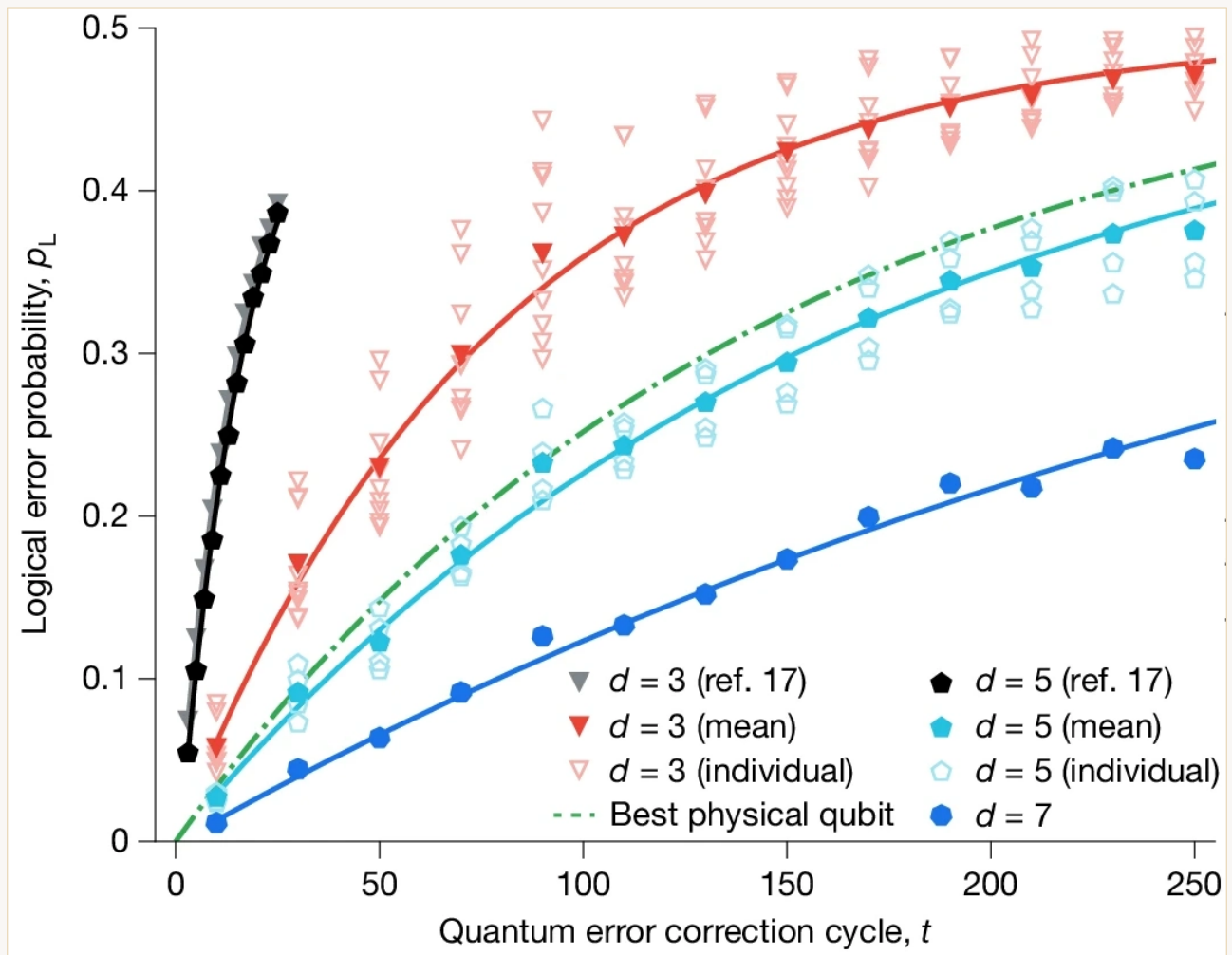
ในเดือนธันวาคม 2024 Google Quantum AI รายงาน การสาธิตอย่างชัดเจนครั้งแรกของอีกระบบหนึ่ง บน Willow โปรเซสเซอร์ตัวนำยิ่งยวดรุ่นล่าสุด ทีมสร้าง หน่วยความจำแบบ surface code ที่ ระยะโค๊ด (code distance) 3, 5 และ 7 และเห็น อัตราข้อผิดพลาดเชิงตรรกะ *ลดลง* ทุกครั้งที่ code ใหญ่ขึ้น — ด้วยปัจจัย $\Lambda = 2.14 \pm 0.02$ ต่อการเพิ่ม distance สองชั้น หน่วยความจำใหญ่ที่สุดแบบระยะ 7 ใช้ 101 คิวบิต เก็บ คิวบิตเชิงตรรกะ ด้วย ข้อผิดพลาด $0.143\% \pm 0.003\%$ ต่อรอบการแก้ไข ข้อผิดพลาด และ — พาดหัวซ้อนในพาดหัว — มันอยู่ได้นานกว่า *คิวบิตกายภาพ ที่ดีที่สุดของตัวเอง* 2.4 ± 0.3 เท่า สิ่งนี้เรียกว่า “ดีกว่าจุดคุ้มทุนของคิวบิตกายภาพ” และเป็นครั้งแรกที่ระบบ การแก้ไขข้อผิดพลาด โดยรวมคืนกำไรให้ตัวเองบนฮาร์ดแวร์นี้

นี่เป็นหมุดหมายจริง และควรแม่นยำว่าเป็นหมุดหมายชนิดใด มันพิสูจน์ว่า scaling ตอนนี้ไปในทิศทางที่ต้องการ ไม่ใช่คอมพิวเตอร์ควอนตัมที่ทำงานครบ และ ตัวงาน ก็ไม่ได้อ้างเช่นนั้น

“surface code”, “distance” และ “ต่ำกว่าเกณฑ์ (below threshold)” หมายถึงอะไร

Logical คิวบิต คือหน่วยข้อมูลควอนตัมที่ได้รับการป้องกันหนึ่งหน่วย ซึ่ง เข้ารหัส กระจายอยู่บน คิวบิตกายภาพ หลายตัว **Surface code** คือวิธีหนึ่งในการ เข้ารหัส บนกริด 2 มิติ โดยมี คิวบิต สำหรับ “วัด” เพิ่มเข้ามาคอยตรวจ ข้อผิดพลาด อย่างต่อเนื่องโดยไม่รบกวนข้อมูลที่เก็บอยู่ **Code distance d** ไม่ใช่ระยะทางจริง แต่คือจำนวน ข้อผิดพลาด ที่วางตำแหน่งเหมาะสมที่สุดซึ่งสามารถทำให้ คิวบิตเชิงตรรกะ เสียหายได้โดย code ตรวจไม่พบ ค่า d ที่ใหญ่ขึ้นหมายถึง patch ที่ใหญ่และทนทานกว่า — ใช้คิวบิตกายภาพ มากขึ้น (ประมาณ $2d^2 - 1$) และแก้ไข ข้อผิดพลาด พร้อมกันได้มากขึ้น สูงสุด $(d - 1)/2$ ดังนั้นสามขนาดที่ทดสอบคือ ระยะ 3, 5 และ 7 สามารถแก้ไข ข้อผิดพลาด พร้อมกันได้ 1, 2 และ 3 จุด และใช้คิวบิตกายภาพ โดยประมาณ 17, 49 และ 97 ตัว — หน่วยความจำ ระยะ 7 ที่ Google สร้างใช้ 101 ตัว มากกว่าค่าต่ำสุดตามตำราเล็กน้อย

คำว่า “ต่ำกว่าเกณฑ์ (below threshold)” คือแก่นสำคัญ Error correction จะช่วยได้ก็ต่อเมื่อ physical อัตราข้อผิดพลาด อยู่ต่ำกว่าค่าวิกฤต; ในบริเวณนั้น ทุกครั้งที่เพิ่ม distance อัตราข้อผิดพลาดเชิงตรรกะ จะถูกกดลงแบบเอ็กซ์โพเนนเชียล ปัจจัยกด ข้อผิดพลาด Λ ใช้อัดสิ่งนี้ — $\Lambda > 1$ หมายความว่าขยาย code แล้วดีขึ้น และยิ่ง Λ สูงยิ่งดี Google รายงาน $\Lambda \approx 2.14$ หมายความว่าทุกการเพิ่ม distance สองชั้น อัตราข้อผิดพลาดเชิงตรรกะ ลดลงประมาณครึ่งหนึ่ง การที่ Λ อยู่เหนือ 1 อย่างชัดเจนคือผลลัพธ์ทั้งเรื่อง



กราฟแสดงว่า logical ข้อผิดพลาด สะสมผ่านรอบการแก้ไข ข้อผิดพลาด อย่างไรสำหรับหน่วยความจำ distance-3, -5 และ -7 (จากบนลงล่าง) เส้นที่ต้องดูคือเส้นประสีเขียว — คิวบิตกายภาพ เดียวที่ดีที่สุด บนชิป หน่วยความจำ ระยะ 7 (สีน้ำเงิน อยู่ล่างสุด) สะสม ข้อผิดพลาด ช้ากว่า เส้นนั้น ดังนั้น เข้ารหัส คิวบิต จึงอยู่ได้นานกว่า คิวบิตกายภาพ ที่ดีที่สุดที่ใช้สร้างมัน — “ดีกว่าจุดคุ้มทุนของคิวบิตกายภาพ” 2.4× นี่คือ ผลด้านอายุการเก็บข้อมูล; ส่วนการกด ข้อผิดพลาด แบบ below-threshold เอง ($\Lambda = 2.14$ เมื่อ code โตจาก ระยะ 3 เป็น 5 เป็น 7) อยู่ใน ตัวเลขของเนื้อหา

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

ผู้วิจัยทำอะไร

- สร้าง หน่วยความจำแบบ surface code บนชิป Willow สองตัว: โปรเซสเซอร์ 105 คิวบิต ใช้รัน ระยะโค้ด (code distance)-3, -5 และ -7 สำหรับการทดสอบ scaling (ใหญ่ที่สุดคือหน่วยความจำ ระยะ 7 ขนาด 101 คิวบิต มี data คิวบิต 49 ตัว) และ โปรเซสเซอร์ 72 คิวบิต ใช้รันหน่วยความจำ distance-5 พร้อม ตัวถอดรหัสแบบเวลาจริง รวมถึง repetition codes ที่มี distance สูง
- วัดว่า logical ข้อผิดพลาด *ต่อรอบ* เปลี่ยนอย่างไรเมื่อเพิ่ม ระยะโค้ด (code distance) จาก 3 เป็น 5 เป็น 7 เพื่อหาปัจจัยกด ข้อผิดพลาด Λ
- เปรียบเทียบอายุของ คิวบิตเชิงตรรกะ กับ คิวบิตกายภาพ เดียวที่ดีที่สุดบนชิปเดียวกัน เพื่อทดสอบ “breakeven”
- รัน ระยะโค้ด (code distance)-5 ด้วย **ตัวถอดรหัสแบบเวลาจริง** — ฮาร์ดแวร์คลาสสิกที่แปลผล ข้อผิดพลาด checks ได้ เร็วกว่าที่มันถูกสร้าง — นานถึงหนึ่งล้านรอบ เพื่อแสดงว่า การแก้ไขข้อผิดพลาด ตามเครื่องนั้น

- ขยาย **repetition codes** ที่ง่ายกว่าไปถึง distance 29 เพื่อค้นหาแหล่ง ข้อผิดพลาด หายากระดับลึกที่ตั้งพื้นประสิทธิภาพ

สิ่งที่พบ

- **Code อยู่ ต่ำกว่าเกณฑ์ (below threshold)** Logical ข้อผิดพลาด ต่อบรรลุด้วย $\Lambda = 2.14 \pm 0.02$ ต่อการเพิ่ม distance สองชั้น — เป็นการทดแทนเชิงทฤษฎีอย่างสะอาดตามที่ทฤษฎีสัญญาไว้ และก่อนหน้านี้ยังไม่มีโปรเซสเซอร์ใดแสดงได้ชัดเจน
- **หน่วยความจำ ระยะ 7 มี ข้อผิดพลาด $0.143\% \pm 0.003\%$ ต่อบรรลุ** และอยู่ได้นานกว่า คิวบิตกายภาพ ที่ดีที่สุด 2.4 ± 0.3 เท่า — ดีกว่าจุดคุ้มทุนของคิวบิตกายภาพ
- **Real-time decoding ตามทัน** Decoder มี latency เฉลี่ย 63 ไมโครวินาทีที่ ระยะ 5 เทียบกับ cycle time 1.1 ไมโครวินาที และรักษาการทำงานได้หนึ่งล้นรอบ — การแก้ไขข้อผิดพลาด ทำงานสด ไม่ใช่วิเคราะห์ย้อนหลังที่หลัง
- **ยังมีแหล่ง ข้อผิดพลาด ลึกที่หายากอยู่** ในการทดสอบ repetition code ประสิทธิภาพสุดท้ายถูกจำกัดด้วย ข้อผิดพลาดที่มีสหสัมพันธ์ bursts ที่เกิดประมาณ **หนึ่งครั้งต่อชั่วโมง** (ราวหนึ่งครั้งในทุก 3×10^9 รอบ) ตั้ง พื้นข้อผิดพลาด ใกล้ 10^{-10} ซึ่งผู้เขียนระบุว่า **ยังไม่เข้าใจต้นเหตุ**

สิ่งที่งานนี้ไม่ได้พิสูจน์

- นี่ **ไม่ใช่** คอมพิวเตอร์ควอนตัมที่กำลังคำนวณ นี่คือ **หน่วยความจำควอนตัม**: เก็บและป้องกัน คิวบิตเชิงตรรกะ หนึ่งตัว ไม่ได้ทำการดำเนินการเชิงตรรกะ (เกต) ระหว่าง คิวบิตเชิงตรรกะ และไม่ได้รันอัลกอริทึม
- นี่ **ไม่ได้** หมายความว่าเหลืออีก คิวบิต เดียวก็ได้เครื่องที่ใช้ประโยชน์ หนึ่งคิวบิตเชิงตรรกะ แบบระยะ 7 ใช้คิวบิตกายภาพราว 101 ตัว; ข้อผิดพลาด ประมาณ 0.1% ต่อบรรลุยังสูงกว่าราว 10^{-6} ถึง 10^{-10} ที่อัลกอริทึมจริงต้องการมาก การปิดช่องว่างต้องไปสู่ distance ที่ใหญ่กว่ามาก — คิวบิตกายภาพ ต่อ คิวบิตเชิงตรรกะ มากขึ้นมาก — และอัลกอริทึมที่มีประโยชน์ต้องใช้คิวบิตเชิงตรรกะ **หลายพันตัว** พร้อมกัน งบ คิวบิตกายภาพ จึงไปถึงระดับหลายล้าน
- คำว่า **“if scaled” มีน้ำหนักจริง** ข้อสรุปของงานเองคือประสิทธิภาพอุปกรณ์ *หาก scale ได้* อาจตอบโต้ภัยอัลกอริทึมขนาดใหญ่ การแสดงว่าแนวโน้มถูกต้องบน คิวบิตเชิงตรรกะ หนึ่งตัวไม่เท่ากับสร้างเครื่องที่ scale แล้ว และไม่มีอะไรที่รับประกันว่าแนวโน้มจะอยู่รอดเมื่อขยายใหญ่กว่านี้มาก
- **Error floor ที่ยังอธิบายไม่ได้เป็นปัญหาจริง** correlated bursts ที่จำกัด repetition-code performance มีขนาดใหญ่กว่าคาดหลายลำดับชั้นตามคำของผู้เขียน และจะขัดขวาง fault-tolerant applications ที่ใหญ่กว่านั้นกว่าจะเข้าใจ — เป็นข้อบกพร่องเปิดที่ระบุไว้ตรง ๆ ไม่ใช่รายละเอียดที่แก้แล้ว
- งานนี้ไม่พูดอะไรเรื่อง การถอดรหัส **encryption** หรือ **“quantum supremacy”** สำหรับงานที่มีประโยชน์ สิ่งเหล่านั้นต้องใช้ fault-tolerant machine เต็มรูปแบบซึ่งงานนี้เป็นเพียงซิลิโคน ไม่ใช่การสาธิต

หลักฐานเชิงแรงแคไหน

- **ข้ออ้างหลักเชิงแรงและสำคัญ** การทำงาน ต่ำกว่าเกณฑ์ (below threshold) พร้อมการทด ข้อผิดพลาด แบบเอ็กซ์โพเนนเชียลชัดเจนข้าม ระยะโค้ด (code distance) สามค่า บวกกับอายุ ดีกว่าจุดคุ้มทุนของคิวบิตกายภาพ และ ตัวถอดรหัสแบบเวลาจริง ที่ใช้งานได้ คือชุดสิ่งที่วงการพยายามไปให้ถึง และสาธิตโดยตรง ไม่ใช่ข้ออนุมาน นี่ไม่ใช่สิ่งประดิษฐ์จาก hype แต่เป็นผลวิศกรรมจริงจากทีมชั้นนำ

- **ผู้เขียนระวังขอบเขตของผล** พวกเขาเรียกมันว่า หน่วยความจำที่ทำงานต่ำกว่าเกณฑ์ ระบบ พื้นข้อผิดพลาด จาก ข้อผิดพลาดที่มีสหสัมพันธ์ ที่ยังอธิบายไม่ได้เอง และผูกอนาคตไว้กับคำว่า “if scaled” ที่เด่นชัด การกล่าวเกินจริงมักเกิดในข่าวรอบข้างที่ปิด “หน่วยความจำ คิวบิตเชิงตรรกะ ที่แก้ ข้อผิดพลาด ได้ดีขึ้นเมื่อโตขึ้น” ให้กลายเป็น “คอมพิวเตอร์ควอนตัมมาแล้ว”
- **สถานะที่ชัดเจนคือพื้นฐานรากที่ทำได้อย่างสะอาด** คิวบิตเชิงตรรกะ หนึ่งตัว ได้รับการป้องกันดีพอจนการเพิ่ม redundancy ช่วยเสียที่ — โดยยังมีเส้นทางยาว ยาก และไม่รับประกันด้าน scaling, เกตเชิงตรรกะ และ ข้อผิดพลาด ที่ยังอธิบายไม่ได้รออยู่

ทำไมจึงสำคัญ

Fault-tolerant คอมพิวเตอร์ควอนตัม มีปัญหาใกล้กับไข่มุตลอด: เครื่องที่มีประโยชน์ต้องการ อัตราข้อผิดพลาด ที่คิวบิตกายภาพ ไม่มีทางทำได้ และทางแก้ — การแก้ไขข้อผิดพลาด — จะทำงานก็ต่อเมื่อฮาร์ดแวร์ดีพอที่จะอยู่ ต่ำกว่าเกณฑ์ (below threshold) ก่อน การข้ามเส้นนั้นได้สักครั้ง แม้มัน คิวบิตเชิงตรรกะ เพียงตัวเดียว เปลี่ยนคำถามจาก “เป็นไปได้จริงไหม?” เป็น “scale ได้ไกลแค่ไหน และเร็วแค่ไหน?” นั่นคือการเปลี่ยนที่จริงและมีความหมาย จึงสมควรได้รับความสนใจ

แต่ความระมัดระวังแบบเดียวที่ทำให้ผลน่าเชื่อถือก็ควรลดอุณหภูมิของเรื่องเล่ารอบมัน นี่คืออริสก่อนแรกของฐานรากที่วางได้ดี ไม่ใช่ตัวอาคาร และคนวางอิฐเองก็เป็นกลุ่มแรกที่บอกเช่นนั้น วิธีติดตาม คอมพิวเตอร์ควอนตัม ในอีกไม่กี่ปีที่ถูกตั้งคือดูเส้นโค้งที่ใหม่หรือหวน: ปัจจัยกีด ข้อผิดพลาด จะคงอยู่เมื่อ code โตหรือไม่, เกตเชิงตรรกะ ทำได้สะอาดเท่า หน่วยความจำเชิงตรรกะ หรือไม่, และ ข้อผิดพลาด ลึกกลับที่มาเพียงชั่วโหม่งละครังจะถูกอธิบายได้หรือไม่

สรุป

Google Quantum AI แสดงอย่างสะอาดเป็นครั้งแรกว่า หน่วยความจำควอนตัม แบบ surface code สามารถทำงาน *ต่ำกว่าเกณฑ์ (below threshold)*: เมื่อขยาย โด่ดจาก ระยะ 3 เป็น 5 เป็น 7 อัตราข้อผิดพลาดเชิงตรรกะ ลดลงแบบเอ็กซ์โพเนนเชียล (ประมาณ $2.14 \times$ ต่อสองขั้น) และหน่วยความจำใหญ่ที่สุดแบบระยะ 7 ขนาด 101 คิวบิต อยู่ได้นานกว่า คิวบิตกายภาพ ที่ดีที่สุด — ดีกว่าจุดคุ้มทุนของคิวบิตกายภาพ — ขณะที่ การแก้ไขข้อผิดพลาด ทำงานแบบ real time นี่เป็นหมุดหมายด้านวิศวกรรมคอมพิวเตอร์ควอนตัมที่ตามหามานานจริง แต่ยังเป็น คิวบิตเชิงตรรกะ เพียงหนึ่งตัวที่ทำหน้าที่เป็นหน่วยความจำ มี อัตราข้อผิดพลาด ห่างจากที่อัลกอริทึมจริงต้องการมาก ไม่ได้ทำ logical operations มี พื้นข้อผิดพลาด ที่ยังอธิบายไม่ได้ซึ่งผู้เขียนระบุเอง และยังต้อง scaling อีกหลายลำดับขนาด เส้น threshold ถูกข้ามจริง — ไม่ใช่ คอมพิวเตอร์ควอนตัมที่ส่งมอบแล้ว

แหล่งข้อมูล

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>