Стабілізаторний формалізм (продовження). Поверхневі коди.

Лекція 11

18 квітня 2023

Стабілізаторний формалізм (продовження)

Нехай ми маємо набір елементів $\{g_1, \ldots, g_k\}$ з групи Паулі на n кубітах.

1. Як перевірити, що підгрупа $G = \langle g_1, \ldots, g_k \rangle$ яку вони генерують, це стабілізаторний код типу $[n, n-k]$, тобто g_i комутують, $-I \notin G$, та g_i незалежні?
2. Як потім знайти додаткові генератори h_i, такі що $\{g_1, \ldots, g_k, (-1)^{b_1} h_1, \ldots, (-1)^{b_{n-k}} h_{n-k}\}$ будуть утворювати код типу $[n, 1]$ (тобто кодувати стабілізаторний стан)?

$-I \notin G$ еквівалентно тому, що $\forall i \; g_i^2 = I$.

Зображення елеменів групи Паулі

Нехай $g \in \Pi^n$. Поставимо у відповідність до g вектор-рядок $b(g)$ довжини $n+n$, що складається із бітів, наступним чином.

Якщо g на i-му кубіті діє як I, то на місцях i та $n+i$ вектору $b(g)$ будуть стояти 0 та 0.
Якщо g на i-му кубіті діє як X, то на місцях i та $n+i$ вектору $b(g)$ будуть стояти 1 та 0.
Дії Z будуть відповідати 0 та 1, а дії Y відповідають $1, 1$.
Фаза генератору неважлива.

Наприклад, якщо

$$g = -iY \otimes X \otimes I \otimes Z \otimes Z,$$

то йому буде відповідати вектор

$$b(g) = \left[1 \quad 1 \quad 0 \quad 0 \quad 0 \quad \middle| \quad 1 \quad 0 \quad 0 \quad 1 \quad 1\right] = \left[b_1(g) \quad \middle| \quad b_2(g)\right].$$

Зображення елеменів групи Паулі

Не важко бачити, що два елементи $g, g' \in \Pi^n$ комутують тоді і лише тоді, коли

$$b(g) \begin{bmatrix} 0 & I_n \\ I_n & 0 \end{bmatrix} b(g')^T = b_1(g) \cdot b_2(g') + b_2(g) \cdot b_1(g') = 0$$

(по модулю 2).

Для набору генераторів $\{g_1, \ldots, g_k\}$ визначають так звану провірочну матрицю, в якій рядки це вектори $b(g_i)$. Наприклад,

$$\left[\begin{array}{ccccccc|ccccccc} 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{array}\right]$$

Провірочна матриця

Теорема

Елементи $\{g_1, \ldots, g_k\}$, які комутують і не породжують $-I$, незалежні тоді й лише тоді, коли вектори $\{b(g_1), \ldots, b(g_k)\}$ лінійно незалежні, тобто провірочна матриця має ранг k (у полі $\mathbb{F}_2$).

Доведення

Не важко перевірити, що $b(g) + b(g') = b(gg')$, тобто додавання відповідних векторів відповідає множенню елементів групи. Виходить, що для лінійної комбінації

$$\sum_i a_i b(g_i) = b(\prod_i g_i^{a_i}).$$

Але ж $b(g) = 0$ тільки якщо $g = \lambda I$, і з умов випливає що $\lambda = 1$. Тож лінійна комбінація може бути нулем тільки якщо $\prod_i g_i^{a_i} = I$, тобто є залежність (для $a_j = 1$ можна виразити $g_j = g_j^{-1}$ через інші).

Стандартна форма стабілізаторного коду

Нехай $\langle g_1, \ldots, g_{n-k} \rangle$ це стабілізаторний код типу $[n, k]$, а G його провірочна матриця розміру $n - k \times 2n$, тобто

$$G = \left[G_1 \quad | \quad G_2 \right].$$

Маємо, що

- Перестановка генераторів (перестановка індексів) відповідає перестановці рядків матриці G;
- Перестановка кубітів відповідає перестановці стовпчиків в кожній з половинок G_1, G_2;
- Заміна генератору g_i на $g_i g_j$, $i \neq j$, відповідає додаванню рядка j до рядка i у матриці G.

Стандартна форма стабілізаторного коду

За допомогою цих перетворень методом Гауса матрицю G можна звести до

$$\begin{array}{r} \\ r\{ \\ n-k-r\{ \end{array} \left[\begin{array}{cccc|cc} \overbrace{I}^{r} & \overbrace{A}^{n-r} & & & \overbrace{B}^{r} & \overbrace{C}^{n-r} \\ 0 & 0 & & & D & E \end{array}\right],$$

де r це ранг G_1. Далі, методом Гауса спрощуємо матрицю E і отримуємо

$$\begin{array}{r} \\ r\{ \\ n-k-r-s\{ \\ s\{ \end{array} \left[\begin{array}{ccccc|ccc} \overbrace{I}^{r} & \overbrace{A_1}^{n-k-r-s} & \overbrace{A_2}^{k+s} & & & \overbrace{B}^{r} & \overbrace{C_1}^{n-k-r-s} & \overbrace{C_2}^{k+s} \\ 0 & 0 & 0 & & & D_1 & I & E_2 \\ 0 & 0 & 0 & & & D_2 & 0 & 0 \end{array}\right].$$

Останні s генераторів мають комутувати з першими r, але це можливо лише якщо $D_2 = 0$, $s = 0$.
Також ми можемо звести C_1 до 0 додаванням рядків.

Стандартна форма стабілізаторного коду

Загалом виходить, що G можна звести до наступної форми:

$$\begin{array}{r} \\ r\{ \\ n-k-r\{ \end{array} \left[\begin{array}{ccc|ccc} \overbrace{I}^{r} & \overbrace{A_1}^{n-k-r} & \overbrace{A_2}^{k} & \overbrace{B}^{r} & \overbrace{0}^{n-k-r} & \overbrace{C}^{k} \\ 0 & 0 & 0 & D & I & E \end{array}\right]$$

Будь-яку таку форму провірочної матриці називають стандартною формою (взагалі кажучи, вона не унікальна).

По цій формі зручно визначати додаткові генератори h_i, що комутують між собою, усіма g_i та разом утворюють незалежну систему. Наприклад, h_i можна взяти такі, що їх провірочна матриця H буде мати вигляд

$$H = \left[\begin{array}{ccc|ccc} 0 & 0 & 0 & A_2^T & 0 & I \end{array}\right],$$

де розміри блоків відповідають розмірам блоків у стандартній формі G.

Логічні кубіти та операції

Як зазначалося на попередній лекції, додаткові h_i дозволяють визначити кодування логічних кубітів стабілізаторного підпростору:

$$|b_1 b_2 \dots b_k\rangle_L \leftrightarrow \langle g_1, \dots, g_{n-k}, (-1)^{b_1} h_1, \dots, (-1)^{b_k} h_k \rangle$$

Більше того, при цьому генератори h_i можна ототожнити з логічними $\bar{Z}_i$:

$$\bar{Z}_i \, |b_1 b_2 \dots b_k\rangle_L = (-1)^{b_i} \, |b_1 b_2 \dots b_k\rangle_L \, .$$

Логічним $\bar{X}_j$ будуть відповідати оператори $f_j \in \Pi^n$, що $f_j h_j = -h_j f_j$ та f_j комутують з іншими генераторами коду. В попередньому прикладі їх можна отримати з матриці

$$F = \left[\begin{array}{ccc|ccc} 0 & E^T & I & C^T & 0 & 0 \end{array}\right].$$

Логічні кубіти та операції

Знання логічних $\bar{Z}_i$ та $\bar{X}_i$ достатньо для того, щоб визначити будь-який інший логічний оператор. Адже $\bar{Y}_i = -i\bar{X}_i\bar{Z}_i$, а всі оператори виду

$$M_1 \otimes \cdots \otimes M_k$$

де кожне M_i це одне з I_i, $\bar{X}_i$, $\bar{Y}_i$, $\bar{Z}_i$, утворють базис усього простору лінійних операторів (розмірності 4^k), тобто будь-який оператор це їх лінійна комбінація.

Наприклад, логічний CNOT можна визначити із рівності

$$\mathrm{CNOT} = \frac{1}{2}(I + Z) \otimes I + \frac{1}{2}(I - Z) \otimes X.$$

Як краще реалізувати логічні операції залежить від конкретних кодів.

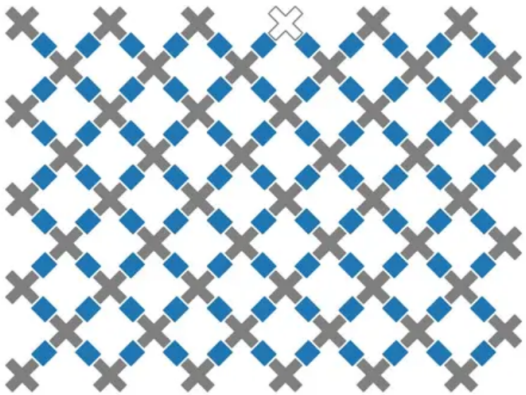
Топологія зв'язків між кубітами.

Для реалізації будь-якого унітарного перетворення на кубітах достатньо вміти застосовувати однокобутні перетворення та двокубітні $\mathrm{CNOT}_{k,l}$. Але двокубітні $\mathrm{CNOT}_{k,l}$ насправді потрібні не між усіма кубітами, деякі можна виразити через інші. Наприклад,

$$\mathrm{CNOT}_{1,3} = \mathrm{SWAP}_{1,2} \cdot \mathrm{CNOT}_{2,3} \cdot \mathrm{SWAP}_{1,2}$$

Зв'язкі між кубітами на квантовому комп'ютері (тобто пари кубітів, де ми можемо застосовувати CNOT) утворюють граф, який називають топологією квантового комп'ютера.

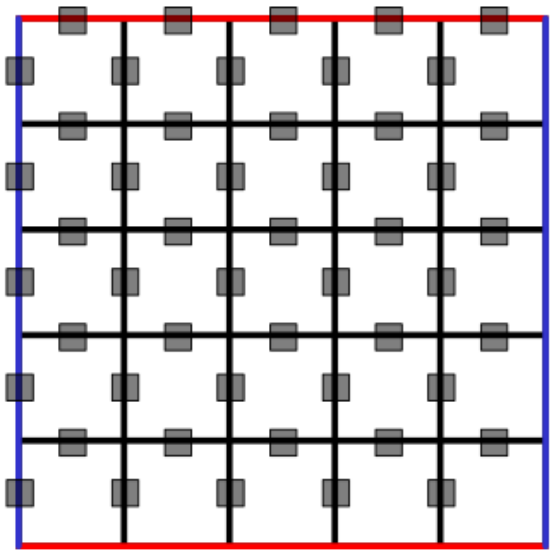
Необхідною умовою є зв'язність цього графа. Зазвичай, кубіти розташовують на площині та з'єднують відповідно до планарної решітки.
Топологія 53-х кубітного Google Sycamore:

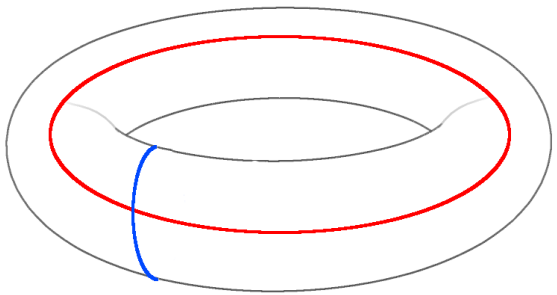


Поверхневі коди. Торичний код.

Поверхневими називають клас кодів корекції квантових похибок, які для кодування використовують набір кубітів, зв'язки між якими дозволяють їх розташувати на якійсь поверхні.

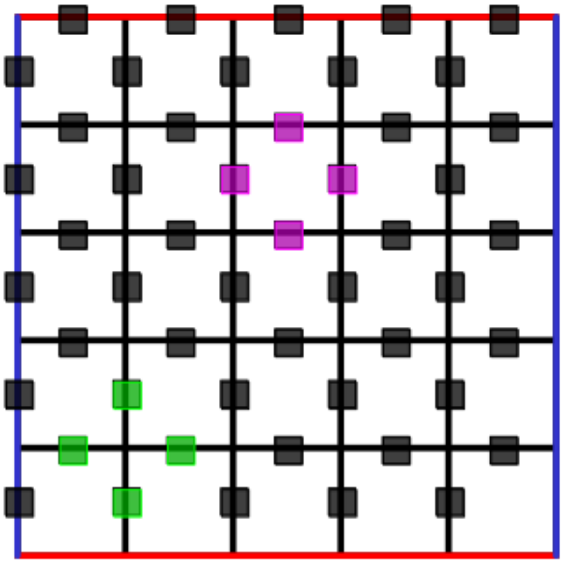
Найпершим прикладом був торичний код А. Кітаєва (1997). В цьому коді $2n^2$ кубітів розташовані на сторонах квадратної решітки, яка циклічна по обох осях, тобто можна вважати, що вона лежить на поверхні тора.





Торичний код

Торичний код є стабілізаторним. В якості генераторів береться n^2 операторів A_s та n^2 операторів B_t:

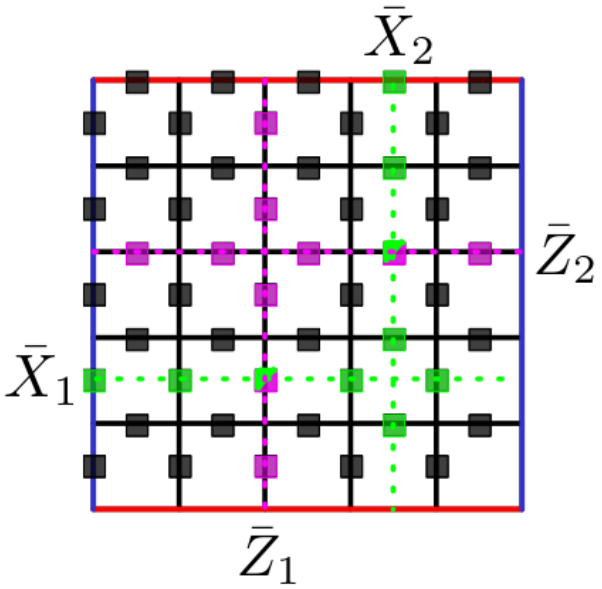


$$A_s = \prod_{i \in \mathrm{star}(s)} X_i = X_{i_1} \otimes X_{i_2} \otimes X_{i_3} \otimes X_{i_4},$$

$$B_t = \prod_{j \in \mathrm{tile}(t)} Z_j = Z_{j_1} \otimes Z_{j_2} \otimes Z_{j_3} \otimes Z_{j_4}.$$

Всі A_s, B_t комутують між собою і не породжують $-I$. Вони є залежні бо $\prod_s A_s = I$ та $\prod_t B_t = I$. Але набір стане незалежним якщо викинути один з генераторів A_s та один з генератор B_t. Це виходить стабілізаторний код типу $[2n^2, 2]$, тобто він кодує 2 логічні кубіти.

Торичний код



Через $\bar{Z}_1$ позначимо добуток Z_j по одній з вертикалей, а через $\bar{Z}_2$ добуток Z_j по одній з горизонталей. Аналогічно, $\bar{X}_1$ добуток X_i по одній з горизонталей, а $\bar{X}_2$ це добуток Z_j по одній з вертикалей.

Оператори $\bar{Z}_1$, $\bar{Z}_2$ комутують між собою і з усіма A_s, B_t. Разом вони задають кодування логічних кубітів:

$$|00\rangle_L \leftrightarrow \langle \{A_s\}', \{B_t\}', \bar{Z}_1, \bar{Z}_2 \rangle,$$

$$|01\rangle_L \leftrightarrow \langle \{A_s\}', \{B_t\}', \bar{Z}_1, -\bar{Z}_2 \rangle,$$

$$|10\rangle_L \leftrightarrow \langle \{A_s\}', \{B_t\}', -\bar{Z}_1, \bar{Z}_2 \rangle,$$

$$|11\rangle_L \leftrightarrow \langle \{A_s\}', \{B_t\}', -\bar{Z}_1, -\bar{Z}_2 \rangle,$$

Торичний код. Виправлення похибок.

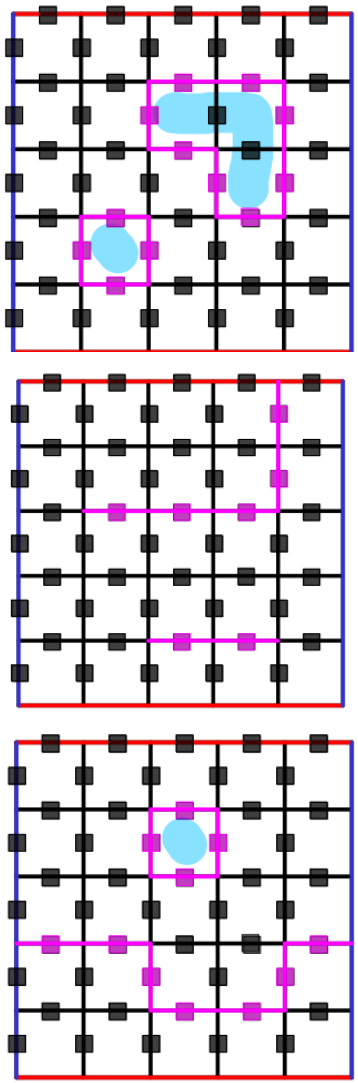
Будемо казати, що стандартна похибка - це операції Z на якійсь підмножині кубітів, разом із X на якійсь підмножині кубітів. Всі інші похибки (унітарні оператори) це лінійні комбінації стандартних похибок. Дія стандартної похибки E на торичний код змінить його на код

$$\langle \{EA_sE^\dagger\}', \{EB_tE^\dagger\}' \rangle = \langle \{\pm A_s\}', \{\pm B_t\}' \rangle$$

де -1 перед генератором з'явиться якщо E антикомутувала з ним, $+1$ залишиться у випадку комутації.

Як і в будь-якому стабілізаторному коді, синдромні вимірювання відповідають операторам $\{A_s\}', \{B_t\}'$. Разом вони дадуть відповідний набір ± 1.

Торичний код. Типи Z похибок.



Тривіальні, які не відстежуються, але і не змінюють стан логічних кубітів. Відповідають границі замкненої області.

Відстежувані, тобто які покажуть нетривіальні синдромні вимірювання.

Невідстежувані, які призводять до зміни логічного стану.
Відповідають лінії, що проходить крізь тор.

Посилання

Стабілізаторний формалізм
https://arxiv.org/abs/quant-ph/9705052

Торичний код
https://arxiv.org/abs/quant-ph/9707021
https://www.youtube.com/watch?v=M25fBmF9XR0

Завдання

1. Довести, що $b(g) + b(g') = b(gg')$ для будь-яких g, g'.
2. Нехай на торичному коді з $2n^2$ кубітів сталася помилка Y рівно на одному з кубітів. Скільки синдромних вимірювань серед $\{A_s\}, \{B_t\}$ (тобто $2n^2$ штук) покажуть -1?