Clifford group

Maris Ozols

July 31, 2008

Abstract

This is a survey on the Clifford group on n qubits. I will discuss its properties and applications in quantum computing.

1 Pauli matrices

The *Pauli matrices* on a single qubit are $I = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$, $X = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$, $Y = \left(\begin{smallmatrix} 0 & -i \\ i & 0 \end{smallmatrix}\right)$, $Z = \left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right)$. On n qubits the set of Pauli matrices is $P_n = \{\sigma_1 \otimes \cdots \otimes \sigma_n \,|\, \sigma_i \in \{I, X, Y, Z\}\}$, $|P_n| = 4^n$. The group $P_n/U(1)$ is isomorphic to a vector space over $\mathbb{F}_2$ with dimension $2n$ via identification:

$$\begin{array}{ccc} Z & \text{—} & Y \\ | & & | \\ I & \text{—} & X \end{array} \quad \Longleftrightarrow \quad \begin{array}{ccc} (0,1) & \text{—} & (1,1) \\ | & & | \\ (0,0) & \text{—} & (1,0) \end{array} \tag{1}$$

where the multiplication of matrices corresponds to the addition of vectors.

2 Clifford group

2.1 Definition

To define the Clifford group, we do not have to turn the Pauli matrices into a group. We need just non-identity Pauli matrices $P_n^* = P_n \setminus \{I^{\otimes n}\}$ (their eigenvalues are ± 1 with equal multiplicity). We can ignore the global phase, since U and $e^{i\varphi}U$ act in the same way:

Definition. The *Clifford group* $\mathcal{C}_n$ on n qubits is

$$\mathcal{C}_n = \left\{ U \in U(2^n) \,|\, \sigma \in \pm P_n^* \Rightarrow U \sigma U^\dagger \in \pm P_n^* \right\} / U(1). \tag{2}$$

2.2 Single qubit case

We have $\pm P_1^* = \{\pm X, \pm Y, \pm Z\}$. Conjugation must preserve the structure of P_1^*, thus the action of $U \in \mathcal{C}_1$ is completely determined by the images of X and Z. Moreover, $UXU^\dagger$ and $UZU^\dagger$ must anti-commute. Thus X can go to any element of $\pm P_1^*$, but Z can only go to $\pm P_1^* \setminus \left\{ \pm UXU^\dagger \right\}$. Hence $|\mathcal{C}_1| = 6 \cdot 4 = 24$.

We can think of $\mathcal{C}_1$ as rotations of the Bloch sphere that permute $\pm x$, $\pm y$, and $\pm z$ directions. There are 6 possibilities where the x axis can go. Once we have fixed the x axis, we can still rotate around it and thus there are 4 possibilities where the z axis can go. $\mathcal{C}_1$ corresponds to the group of rotational symmetries of the cube.

2.3 Number of elements

To fix an element $U \in \mathcal{C}_n$, it is enough to specify how it transforms X_i and Z_i for all $i \in \{1, \ldots, n\}$, since they form a basis of the vector space (1). All X's and Z's commute, except X_i and Z_i that anti-commute (elements that anti-commute are joined by edges):

$$\begin{array}{ccccc} X_1 & X_2 & \ldots & X_{n-1} & X_n \\ | & | & & | & | \\ Z_1 & Z_2 & \ldots & Z_{n-1} & Z_n \end{array} \tag{3}$$

Conjugation by U certainly must preserve this structure. Moreover, it can be shown that there are no other restrictions, i.e., each mapping that sends (3) to distinct elements of $\pm P_n^*$ and preserves their structure, determines a unique $U \in \mathcal{C}_n$. Let us find all such mappings.

What are the possible images of the last pair (X_n, Z_n)? X_n can go to any element of $\pm P_n^*$, but Z_n can only go to elements of $\pm P_n^*$ that anti-commute with $UX_nU^\dagger$. Thus there are $|\pm P_n^*| = 2(4^n - 1)$ choices for X_n. Observe that each matrix in $\pm P_n^*$ anti-commutes with exactly half[1] of Pauli matrices P_n (this half is clearly in P_n^*). Thus, there are $2\,|P_n|\,/2 = 4^n$ choices for Z_n. The elements of $\mathcal{C}_n$ that leave both X_n and Z_n fixed form a group isomorphic to $\mathcal{C}_{n-1}$ with the number of cosets equal to $2(4^n - 1)4^n$. Hence $|\mathcal{C}_n| = 2(4^n - 1)4^n\,|\mathcal{C}_{n-1}|$. Therefore,

$$|\mathcal{C}_n| = \prod_{j=1}^{n} 2(4^j - 1)4^j = 2^{n^2+2n} \prod_{j=1}^{n} (4^j - 1). \tag{4}$$

The first few values of $|\mathcal{C}_n|$ are given in Table 1. Equation (4) does not agree with [1], since in [1] it is assumed that $H, P \in \mathcal{C}_n$, i.e., $\mathcal{C}_n$ is implicitly defined as the group generated by H, P, and $CNOT$ (see the next section) without ignoring the global phase. Since $(PH)^3 = e^{\frac{2\pi i}{8}} I$, this way each Clifford group operation is obtained 8 times with different global phases.

n	1	2	3	4	5
$\|\mathcal{C}_n\|$	24	11520	92897280	12128668876800	25410822678459187200

Table 1: The order the Clifford group $\mathcal{C}_n$ on n qubits ($\frac{1}{8}$ "Sloane's A003956").

2.4 Generators

Theorem. $\mathcal{C}_n = \langle H_i, P_i, CNOT_{ij} \rangle / U(1)$, where

$$H = \frac{1}{\sqrt{2}} \left(\begin{smallmatrix} 1 & 1 \\ 1 & -1 \end{smallmatrix}\right), \quad P = \left(\begin{smallmatrix} 1 & 0 \\ 0 & i \end{smallmatrix}\right), \quad CNOT = \left(\begin{smallmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{smallmatrix}\right). \tag{5}$$

[1] Let $\sigma \in \pm P_n^*$. Let k be a position where σ does not contain identity I. Then all Pauli matrices that anti-commute with σ can be constructed as follows: put any of I, X, Y, Z at each position other than k; then fill the kth position in any of two possible ways so that the obtained matrix anti-commutes with σ.

See [2, p. 13] or [3, Section 5.8] for the proof. Note that for $n = 1$ we need only H and P. It can be easily verified that they generate the rotational symmetry group of a cube. Then we have to use induction on n. The main idea is to show that any Clifford operation on $n + 1$ qubits can be implemented using only those on at most n qubits.

3 Applications

3.1 Gottesman - Knill theorem

Quantum circuits that involve only Clifford group operations are not universal for quantum computing. In fact, one can efficiently simulate such circuits on a classical computer.

Theorem. Any quantum computation involving only:

- state preparation in the computational basis,
- Clifford group operations,
- measurements in the standard basis,
- any classical control conditioned on the measurement outcomes

can be perfectly simulated in polynomial time on a probabilistic classical computer.

The main idea is to use the *stabilizer formalism* to see how the stabilizer of the quantum state evolves instead of following the evolution of the state directly [4]. Aaronson and Gottesman have written a program in C called `CHP` that can simulate such circuits [5]. It can easily handle up to 3000 qubits!

3.2 Universal set of quantum gates

Assume we can implement all operations in $\mathcal{C}_n$ (e.g., it means that we can permute qubits in arbitrary way). If we could implement any other fixed gate, that is not (a multiple of a gate) in $\mathcal{C}_n$, we could apply it on any ordered tuple of qubits. It turns out that this would allow us to perform any quantum computation (Theorem 6.5 in [6]):

Theorem. $\mathcal{C}_n$ together with any other gate not in $\mathcal{C}_n$ form a universal set of quantum gates.

Unfortunately, there is no *elementary* proof available for this theorem. However, for $n = 1$ it is not that hard to prove it. Recall the geometrical meaning of $\mathcal{C}_1$ discussed in Sect. 2.2 – it is the rotational symmetry group of a cube. A classical result says that all finite groups of rotations in $\mathbb{R}^3$ that do not have an invariant 2-dimensional subspace are rotational symmetry groups of Platonic solids. Since there is no other Platonic solid, whose rotational symmetry group properly contains that of a cube, the group obtained by adding any gate to $\mathcal{C}_1$ must be infinite. Moreover, it can be shown that it is dense in $O(3)$ [7].

References

[1] Calderbank R.A., Rains E.M., Shor P.W., Sloane N.J.A., Quantum Error Correction Via Codes Over GF(4), `arXiv:quant-ph/9608006v5`.

[2] Gottesman D., A Theory of fault-tolerant quantum computation, `arXiv:quant-ph/9702029v2`.

[3] Gottesman D., Stabilizer Codes and Quantum Error Correction, PhD thesis, `arXiv:quant-ph/9705052v1`.

[4] Gottesman D., The Heisenberg Representation of Quantum Computers, `arXiv:quant-ph/9807006v1`.

[5] Aaronson S., Gottesman D., Improved Simulation of Stabilizer Circuits, `arXiv:quant-ph/0406196v5`.

[6] Nebe G., Rains E.M., Sloane N.J.A., The Invariants of the Clifford Groups, `arXiv:math/0001038v2`.

[7] Ozols M., Mancinska L., Childs A., Leung D., Universal Pairs of Rotations, available here.