A Complete Set of Transformation Rules for Quantum Boolean Circuits with CNOT gates

Kazuo Iwama*† and Shigeru Yamashita*‡
*Quantum Computation and Information, ERATO, Japan Science and Technology Corporation (JST)
†School of Infomatics, Kyoto University, Kyoto, Japan
‡NTT Communication Science Laboratories, Kyoto, Japan

For given Boolean formulas or circuits, S_1 and S_2, to determine whether or not S_1 is equivalent to S_2 is a fundamental coNP-complete problem. Note that coNP-completeness does not always deny the existence of short *witnesses* or *proofs*. In fact, there have been a lot of efforts to seek so-called *proof systems* which provide us (if any) with a short *proof*. Here a proof is a certificate with which we can verify $S_1 = S_2$ in polynomial time. Such systems include Resolution and the more powerful Frege systems both of which are used to prove that a given CNF formula is unsatisfiable (i.e., for the case that S_1 is a CNF formula and S_2 is the empty formula). Although research efforts for these proof systems have been mostly devoted to obtain (exponential) lower-bounds in the theory community, their original motivation is obviously more positive, namely, to use them for automated theorem proving and circuit design. For example, DeMorgan's law is used very often to make a local simplification of Boolean circuits. Our motivation of this research is to develop a similar proof system for *quantum* Boolean circuits with CNOT gates (generalized Toffoli gates).

In our model of a quantum Boolean circuit with CNOT gates, the input and the corresponding output states of the first n qubits have to be identical. The $(n+1)$-st qubit state $|x_{n+1}\rangle$ is changed to $|x_{n+1} \oplus f(x_1, \cdots, x_n)\rangle$, and is considered as a special bit, so called a *work bit*, which is used to obtain the value of the Boolean function f for inputs $x_1, \cdots, x_n$. Furthermore, a circuit can use any (finite) number of *auxiliary qubits* which are reset to $|0\rangle$ initially. Note that our model can be used for quantum Boolean oracles, which are essential for many useful quantum algorithms. More formally, a *quantum Boolean circuit* is given as a sequence $[t_1, C_1] \cdot [t_2, C_2] \cdots [t_m, C_m]$, where (i) t_i is an integer ≥ 1 and (ii) C_i is a finite set of integers ≥ 1. Each $[t_i, C_i]$ is a CNOT gate whose control bits are given by C_i and its target bit is given by t_i.

The followings are the *transformation rules* of our proof system, which can be applied for a sequence of CNOT gates. Each transformation rule looks like $F \Leftrightarrow G$, which means that we can transform F to G, and vice versa, where F and G are sequences of CNOT gates.

Transformation Rule Set. In the followings, ϵ means the empty sequence, and we refer to a CNOT gate whose target bit is the i-th bit as $CNOT_i$.

(1) $[t_1, C_1] \cdot [t_1, C_1] \Leftrightarrow \epsilon$.

(2) $[t_1, C_1] \cdot [t_2, C_2] \Leftrightarrow [t_2, C_2] \cdot [t_1, C_2]$, if $t_1 \notin C_2$ and $t_2 \notin C_1$. (The condition means that the two gates are "independent." If there is some influence between two gates, we cannot change the order of the two gates. Even in such cases, we can change the order by adding some gates, as we will see in the following Transformation Rules.)

(3) $[t_1, C_1] \cdot [t_2, C_2] \Leftrightarrow [t_2, C_2] \cdot [t_1, C_1] \cdot [t_1, C_1 \cup C_2 - \{t_2\}]$, if $t_1 \notin C_2$ and $t_2 \in C_1$.

(4) $[t_1, C_1] \cdot [t_2, C_2] \Leftrightarrow [t_2, C_1 \cup C_2 - \{t_1\}] \cdot [t_2, C_2] \cdot [t_1, C_2]$, if $t_1 \in C_2$ and $t_2 \notin C_1$. (This rule is a dual case of (3), i.e., the relationship between the two gates is just opposite.)

(5) $[t_1, \{c_1\}] \cdot [t_2, C_2 \cup \{c_1\}] \Leftrightarrow [t_1, \{c_1\}] \cdot [t_2, C_2 \cup \{t_1\}]$, if $(t_1 > n+1)$ and there is no $CNOT_{t_1}$ before $[t_1, \{c_1\}]$.

(6) $[t, C] \Leftrightarrow \epsilon$, if there is a integer i such that $i \in C, i > n+1$, and there is no $CNOT_i$ before $[t, C]$.

We prove the completeness of the above transformation rules, i.e., the existence of a sequence of transformations from any circuit to any other equivalent one by introducing a unique canonical form of a quantum Boolean circuits. Since each transformation is bidirectional, we can prove the completeness of the rule set by only showing that there is a sequence of transformations from any circuit to its canonical form.

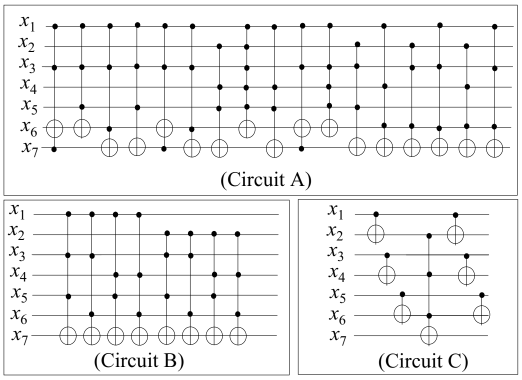

FIG. 1. An Example of Equivalent Circuits

Fig. 1 shows three quantum circuits which are all equivalent, i.e., they compute the same Boolean fucntion $f(x_1, \cdots, x_6) = (x_1 \oplus x_2) \cdot (x_3 \oplus x_4) \cdot (x_5 \oplus x_6)$. Note that x_7 is a work bit in these circuits. We show that we can always construct a circuit by using only generalized Toffoli gates whose target bit is x_{n+1} like Circuit B, and the form is our unique canonical form. However, different types of circuits are of course possible like Circuits A and C, where C is apparently simpler than A and B. By using our transformation rules, we can modify a given circuit into another (e.g., Circuit A to C) with a desirable property (e.g., of small size) by executing a NP-type search, although the length of the search path might not be too short.

The authors are grateful to Mr. Hirotada Kobayashi for his useful comments on the properness of our circuit model.