On the Fault Testing for Reversible Circuits

Satoshi Tayu, Shigeru Ito, and Shuichi Ueno

Department of Communications and Integrated Systems
Tokyo Institute of Technology, Tokyo 152-8550-S3-57, Japan

Abstract. This paper shows that it is NP-hard to generate a minimum complete test set for stuck-at faults on the wires of a reversible circuit. We also show non-trivial lower bounds for the size of a minimum complete test set.

Keywords: 3SAT, CNOT gate, complete test set, NP-complete, stuck-at-fault.

1 Introduction

Reversible circuits, which permute the set of input vectors, have potential applications in nanocomputing [3], low power design [1], digital signal processing [6], and quantum computing [4]. This paper shows that given a reversible circuit C, it is NP-hard to generate a minimum complete test set for stuck-at faults on the wires of C. This is the first result on the complexity of fault testing for reversible circuits, as far as the authors know. We also show non-trivial lower bounds for the size of a minimum complete test set.

A gate is *reversible* if the Boolean function it computes is bijective. If a reversible gate has k input and output wires, it is called a $k \times k$ *gate*. A circuit is *reversible* if all gates are reversible and are interconnected without funout or feedback. If a reversible circuit has n input and output wires, it is called an $n \times n$ *circuit*.

We shall focus our attention to detecting faults in a reversible circuit C which cause wires to be stuck-at-0 or stuck-at-1. Let $W(C)$ be the set of all wires of C. $W(C)$ consists of all output wires of C and input wires to the gates in C. $W(C)$ is the set of all possible fault locations in C. For an $n \times n$ reversible circuit C, a test is an input vector in $\{0,1\}^n$. A test set is said to be *complete* for C if it can detect all possible single and multiple stuck-at faults on $W(C)$. Patel, Hayes, and Markov [5] showed that for any reversible circuit C, there exists a complete test set for C. Let $\tau(C)$ be the minimum cardinality of a complete test set for C.

We first show that it is NP-hard to compute $\tau(C)$ for a given reversible circuit C. Let MTS (Minimum Test Size) be a problem of deciding if $\tau(C) \leq B$ for a given reversible circuit C and integer B. We show in Section 2 that MTS is NP-complete.

Patel, Heyes, and Markov [5] show a surprising upper bound for $\tau(C)$. They showed that

$$\tau(C) \quad = \quad O(\log |W(C)|) \tag{1}$$

for any reversible circuit C. We show the first non-trivial existential lower bound for $\tau(C)$. We show in Section 4 that there exists a reversible circuit C such that

$$\tau(C) \quad = \quad \Omega(\log \log |W(C)|). \tag{2}$$

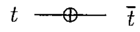
A k-CNOT gate is a $(k+1) \times (k+1)$ reversible gate. It passes some k inputs, referred to as control bits, to the outputs unchanged, and inverts the remaining input, referred to as target bit, if the control bits are all 1. The 0-CNOT gate is just an ordinary NOT gate. A CNOT gate is a k-CNOT gate for some k. Some CNOT gates are shown in Fig. 1, where a control bit and target bit are denoted by a black dot and ring-sum, respectively. A *CNOT circuit* is a reversible circuit consisting of only CNOT gates. A *k-CNOT circuit* is a CNOT circuit consisting of only k-CNOT gates. Any Boolean function can be implemented by a CNOT circuit since the 2-CNOT gate can implement the NAND function.

Chakraborty [2] showed that

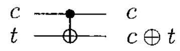
$$\tau(C) \quad \leq \quad n \tag{3}$$

if C is an $n \times n$ CNOT circuit with no 0-CNOT or 1-CNOT gate. We show in Section 5 that there exists an $n \times n$ 2-CNOT circuit C such that

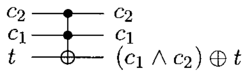
$$\tau(C) = \Omega(\log n). \tag{4}$$



(a) 0-CNOT gate.



(b) 1-CNOT gate.



(c) 2-CNOT gate.

Figure 1: CNOT gates.

It is an interesting open problem to close the exponential gaps between the upper bounds (1) and (3), and our lower bounds (2) and (4), respectively.

2 Complete Test Sets

A wire w of a reversible circuit C is said to be *controllable* by a test set T if the value of w can be set to both 0 and 1 by T. A set of wires $S \subseteq W(C)$ is said to be *controllable* by T if each wire of S is controllable by T. The following characterization for a complete test set is shown in [5].

Theorem I *A test set T for a reversible circuit C is complete if and only if $W(C)$ is controllable by T.* ■

3 NP-Completeness of MTS

The purpose of this section is to prove the following:

Theorem 1 *MTS is NP-complete.*

Proof. A minimum complete test set T for a reversible circuit C can be verified in polynomial time, since $|T| = O(\log|W(C)|)$ by (3). Thus MTS is in NP.

We show a polynomial time reduction from 3SAT, a well-known NP-complete problem, to MTS. Let $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ and

$$\phi(\boldsymbol{x}) = \bigwedge_{j=1}^{m} \rho_j$$

be a Boolean function in conjunctive normal form in which each clause ρ_j has 3 literals for $j \in [m] = \{1, 2, \ldots, m\}$. For a Boolean variable x, literals $\overline{x}$ and x are denoted by x^0 and x^1, respectively.

We use generalized CNOT gates for simplicity. A *generalized k-CNOT gate* has k control bits $x_1, \ldots, x_k$ and a target bit t. The output of the target bit is defined as

$$(x_1^{\alpha_1} \wedge x_2^{\alpha_2} \wedge \cdots \wedge x_k^{\alpha_k}) \oplus t.$$

A control bit x_i is said to be positive if $\alpha_i = 1$, and negative if $\alpha_i = 0$. Notice that a CNOT gate is a generalized CNOT gate with no negative control bit. Notice also that a negative control bit is equivalent to a positive control bit with a 0-CNOT gate on the input and output wires. A *generalized CNOT* [*k-CNOT*] *circuit* is a reversible circuit consisting of only generalized CNOT [k-CNOT] gates.

We first construct a generalized CNOT gate G_j for each clause ρ_j. Let

$$\rho_j = x_{j1}^{\sigma_{j1}} \vee x_{j2}^{\sigma_{j2}} \vee x_{j3}^{\sigma_{j3}},$$

where $\sigma_{jl} \in \{0, 1\}$ and $x_{jl} \in \{x_i | i \in [n]\}$ for $l \in [3]$. We construct a generalized 3-CNOT gate G_j for ρ_j as follows. The gate G_j has 3 control bits x_{j1}, x_{j2}, x_{j3}, and a target bit t. A control bit x_{jl} is defined to be positive if $\sigma_{jl} = 0$, and negative if $\sigma_{jl} = 1$. For an $n \times n$ circuit C and an input vector $\boldsymbol{v} \in \{0, 1\}^n$, we denote by $C(\boldsymbol{v})$ the output vector of C for $\boldsymbol{v}$. If $I \subseteq \{0, 1\}^n$, we define that $C(I) = \{C(\boldsymbol{v}) | \boldsymbol{v} \in I\}$. The following lemma is immediate from the definition of G_j.

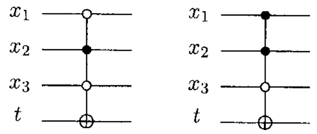
Lemma 1
$G_j(x_{j1}, x_{j2}, x_{j3}, t) = (x_{j1}, x_{j2}, x_{j3}, \overline{\rho_j} \oplus t)$. ■

Lemma 1 means that G_j changes the target bit t for input vector $(x_{j1}, x_{j2}, x_{j3}, t)$ if and only if $\rho_j(x_{j1}, x_{j2}, x_{j3}) = 0$. As an example, for a Boolean function:

$$\left.\begin{array}{rcl} \psi(x_1, x_2, x_3) & = & \rho_1 \wedge \rho_2, \\ \rho_1 & = & x_1 \vee \overline{x_2} \vee x_3, \text{ and} \\ \rho_2 & = & \overline{x_1} \vee \overline{x_2} \vee x_3, \end{array}\right\} \quad (5)$$

generalized 3-CNOT gates G_1 and G_2 are shown in Fig. 2, where a negative control bit is denoted by an empty circle.

We next construct a $(2n+1) \times (2n+1)$ generalized 6-CNOT circuit $C(\phi)$ for ϕ. For $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in \{0, 1\}^n$, $\boldsymbol{y} = (y_1, y_2, \ldots, y_n) \in \{0, 1\}^n$, and $t \in \{0, 1\}$, let $(\boldsymbol{x}, \boldsymbol{y}, t) = (x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n, t)$. Let G'_j be a copy of G_j with control bits $x'_{j1}, x'_{j2}, x'_{j3}$, and a target bit t for any $j \in \{1, 2, \ldots, m\}$. For any $j, h \in \{1, 2, \ldots, m\}$, G_{jh} is a generalized 6-CNOT gate with control bits x_{j1},



(a) G_1.   (b) G_2.

Figure 2: Generalized 3-CNOT gates G_1 and G_2.

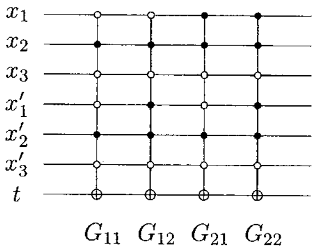


Figure 3: 6-CNOT circuit $C(\psi)$.

$x_{j2}, x_{j3}, x'_{h1}, x'_{h2}, x'_{h3}$, and a target bit t. A control bit $x_{jl}[x'_{hl}]$ is positive in G_{jh} if and only if $x_{jl}[x'_{hl}]$ is positive in $G_j[G'_h]$. We construct a $(2n+1)\times(2n+1)$ generalized 6-CNOT circuit $C(\phi)$ which is a cascade consisting of m^2 gates G_{jh} $(j,h \in [m])$. As an example, $C(\psi)$ for the Boolean function ψ defined in (5) is shown in Fig. 3. We have the following by Lemma 1.

Lemma 2
$G_{jh}((\boldsymbol{x},\boldsymbol{x}',t)) = \left(\boldsymbol{x},\boldsymbol{x}',\left(\overline{\rho_j(\boldsymbol{x})} \wedge \overline{\rho_h(\boldsymbol{x}')}\right) \oplus t\right)$. ■

Lemma 2 implies that G_{jh} changes the target bit if and only if $\rho_j(\boldsymbol{x}) = 0$ and $\rho_h(\boldsymbol{x}') = 0$.

We now show that ϕ is satisfiable if and only if $\tau(C(\phi)) \leq 2$. For a gate G of C, $G(\boldsymbol{v})$ is the output vector of G generated by an input vector $\boldsymbol{v}$ of C. Also, $w(\boldsymbol{v})$ is the value of a wire w generated by $\boldsymbol{v}$.

Lemma 3 *A test set $T = \{\boldsymbol{v}_1, \boldsymbol{v}_2\}$ of a generalized k-CNOT circuit C with $k \geq 2$ is complete if and only if T satsifies the following conditions:*

(i) $\boldsymbol{v}_2 = \overline{\boldsymbol{v}_1}$, *and*

(ii) $G(\boldsymbol{v}_i) = \boldsymbol{v}_i$ $(i \in [2])$ *for every gate G.*

Proof. It is easy to see that if T satisfies (i) and (ii), then $W(C)$ is controllable by T. Thus T is complete for C by Theorem I.

Suppose T is complete for C. Then $W(C)$ is cotrollable by T by Theorem I. Since the input wires of C are controllable by T, we have $\boldsymbol{v}_2 = \overline{\boldsymbol{v}_1}$. Thus, T satisfies (i). Suppose T does not satisfy (ii), that is $G(\boldsymbol{v}_i) \neq \boldsymbol{v}_i$ for some generalized CNOT gate G and some i, say $i = 1$. Since $G(\boldsymbol{v}_1) \neq \boldsymbol{v}_1$ and $\boldsymbol{v}_2 = \overline{\boldsymbol{v}_1}$, we have $G(\boldsymbol{v}_2) = \boldsymbol{v}_2$. If w_{in} and w_{out} are the input and output wires of the target bit of G, we have

$$w_{\text{in}}(\boldsymbol{v}_1) = \overline{w_{\text{out}}(\boldsymbol{v}_1)}.$$

Since $\boldsymbol{v}_2 = \overline{\boldsymbol{v}_1}$ and $G(\boldsymbol{v}_2) = \boldsymbol{v}_2$, we have

$$w_{\text{in}}(\boldsymbol{v}_2) = \overline{w_{\text{in}}(\boldsymbol{v}_1)},$$

and so

$$w_{\text{out}}(\boldsymbol{v}_2) = w_{\text{out}}(\boldsymbol{v}_1),$$

which means that w_{out} is not controllable by T, a contradiction. Thus T satisfies (ii). ■

Now, we are ready to prove the following.

Lemma 4 *ϕ is satisfiable if and only if $\tau(C(\phi)) \leq 2$.*

Proof. It is easy to see from Lemmas 2 and 3 that if $\phi(\boldsymbol{x}) = 1$ for some $\boldsymbol{x} \in \{0,1\}^n$, then a test set $\{(\boldsymbol{x},\overline{\boldsymbol{x}},0),(\overline{\boldsymbol{x}},\boldsymbol{x},1),\}$ is complete for $C(\phi)$. Thus, $\tau(C(\phi)) \leq 2$.

Notice that $\tau(C) \geq 2$ for any reversible circuit C by Theorem I. Suppose $\tau(C(\phi)) = 2$, and let T be a complete test set of size two. By Lemma 3, $T = \{(\boldsymbol{x},\boldsymbol{y},0),(\overline{\boldsymbol{x}},\overline{\boldsymbol{y}},1),\}$ for some $\boldsymbol{x},\boldsymbol{y} \in \{0,1\}^n$. Also by Lemma 3, $G_{jh}((\boldsymbol{x},\boldsymbol{y},0)) = (\boldsymbol{x},\boldsymbol{y},0)$ and $G_{jh}((\overline{\boldsymbol{x}},\overline{\boldsymbol{y}},1)) = (\overline{\boldsymbol{x}},\overline{\boldsymbol{y}},1)$ for any $j,k \in [m]$. Thus by Lemma 2,

$$\overline{\rho_j(\boldsymbol{x})} \wedge \overline{\rho_h(\boldsymbol{y})} = 0 \text{ and } \overline{\rho_j(\overline{\boldsymbol{x}})} \wedge \overline{\rho_h(\overline{\boldsymbol{y}})} = 0$$

for any $j,h \in [m]$, that is,

$$\rho_j(\boldsymbol{x}) \vee \rho_h(\boldsymbol{y}) = 1 \text{ and } \rho_j(\overline{\boldsymbol{x}}) \vee \rho_h(\overline{\boldsymbol{y}}) = 1$$

for any $j,h \in [m]$. If $\rho_j(\boldsymbol{x}) = 1$ for any $j \in [m]$, then $\phi(\boldsymbol{x}) = 1$, and ϕ is satisfiable. If $\rho_j(\boldsymbol{x}) = 0$ for some $j \in [m]$, then $\rho_h(\boldsymbol{y}) = 1$ for any $h \in [m]$. Thus $\phi(\boldsymbol{y}) = 1$, and ϕ is satisfiable. ■

Since $C(\phi)$ can be constructed in polynomial time, we complete the proof of the theorem. ■

4 Lower Bounds for 1-CNOT Circuits

The purpose of this section is to prove the following:

Theorem 2 *There exists a 1-CNOT circuit C such that $\tau(C) = \Omega(\log\log|W(C)|)$.* ■

Before proving the theorem, we need some preliminaries.

4.1 Preliminaries

The level of a wire of a reversible circuit is defined as follows. The input wires of the circuit are at level 0, and the output wires of a gate are at one plus the highest level of any of input wires of the gate. In cases where an input wire of a gate is at level i and the output wires are at level $j > i + 1$, we say the input wire is at all levels between i and $j - 1$ inclusively. It is easy to see the following lemmas.

Lemma 5 *If C_3 is a reversible 2×2 circuit consisting of just one 1-CNOT gate, then $\tau(C_3) = 3$.* ■

Lemma 6 *If B is a 2×2 1-CNOT circuit shown in Fig. 4, then $B(\boldsymbol{v}) = \boldsymbol{v}$ for any $\boldsymbol{v} \in \{0,1\}^2$.* ■

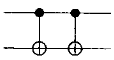


Figure 4: 2×2 1-CNOT circuit B.

Lemma 7 *If C is an $n \times n$ 1-CNOT circuit with g gates, then $|W(C)| = n + 2g$.* ■

4.2 Proof of Theorem 2

We prove the theorem by constructing such circuit. Let C_h ($h \geq 3$) be a 1-CNOT circuit defined as follows. Let C_3 be a 1-CNOT circuit consisting of just one 1-CNOT gate. For $h \geq 4$, C_h is recursively defined as follows. Let $C_{h-1}^{(1)}, C_{h-1}^{(1)}, \ldots, C_{h-1}^{(\varpi_{h-1})}$ be $\varpi_{h-1} + 1$ copies of C_{h-1}, where $\varpi_{h-1} = |W(C_{h-1})|$. Construct an $n_{h-1} \times n_{h-1}$ 1-CNOT circuit D_{h-1} by concatenating $C_{h-1}^{(1)}, C_{h-1}^{(2)}, \ldots, C_{h-1}^{(\varpi_{h-1})}$, where n_{h-1} is the number of input wires of C_{h-1}. Let $W(C_{h-1}^{(k)}) = \{w_1^{(k)}, w_2^{(k)}, \ldots, w_{\varpi_{h-1}}^{(k)}\}$ for $0 \leq k \leq \varpi_{h-1}$ such that if the level of $w_i^{(k)}$ is not greater than the level of $w_j^{(k)}$, then $i \leq j$. C_h is constructed from D_{h-1} and $C_{h-1}^{(0)}$ by inserting two 1-CNOT gates for each wire of $C_{h-1}^{(i)}$, $i \in [\varpi_{h-1}]$, such that the wire of $C_{h-1}^{(i)}$ is the control bit and $w_i^{(0)}$ is the target bit of the both 1-CNOT gates. As an example, D_4 and C_4 are shown in Fig. 5 and Fig. 6, respectively.

From the definition of C_h, we have

$$
\begin{aligned}
n_h &= 2^{h-3}, && (6)\\
g_h &= (|W(C_{h-1})| + 1)g_{h-1} + 2|W(C_{h-1})|^2 \\
&= 10g_{h-1}^2 + g_{h-1}(n_{h-1} + 1) + 2n_{h-1}^2, && (7)
\end{aligned}
$$

where g_h is the number of gates in C_h.

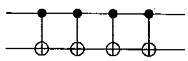


Figure 5: 1-CNOT circuit D_4.

Lemma 8 $h = \Omega(\log\log|W(C_h|))$.

Proof. From (6) and (7), $g_h > n_h$. Thus from (7), $g_h \leq dg_{h-1}^2$ for some constant d, i.e.,

$$
\begin{aligned}
\log g_h &\leq 2(\log g_{h-1} + \log d) - \log d \\
&< 2^{h-3}(\log g_3 + \log d). && (8)
\end{aligned}
$$

Since $|W(C_h)| = n_h + 2g_h \leq 3g_h$ by Lemma 7,

$$
\begin{aligned}
&\log\log|W(C_h)| \\
&\quad \leq \log\log g_h + \log 3 \\
&\quad \leq h - 3 + \log(\log g_3 + \log d) + \log 3
\end{aligned}
$$

by (8), and so $h = \Omega(\log\log|W(C_h)|)$. ■

Lemma 9 $\tau(C_h) \geq h$.

Proof. The proof is by induction on h. $\tau(C_3) = 3$ by Lemma 5. Suppose $\tau(C_{h-1}) \geq h - 1$. We will show that $\tau(C_h) \geq h$. Suppose contrary that $\tau(C_h) = h - 1$, and let $T = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{h-1}\}$ be a complete test set for C_h. Since $\tau(C_{h-1}) \geq h - 1$, $W(C_{h-1})$ is not controllable by $T' = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{h-2}\}$. Thus there exist i and j such that neither $w_i^{(0)}$ nor $w_j^{(i)}$ are controllable by T'. It follows that if G is a 1-CNOT gate with the control bit on $w_j^{(i)}$ of $C_{h-1}^{(i)}$ and the target bit on $w_i^{(0)}$, then $W(G)$ is not controllable by T, a contradiction. Thus, we have $\tau(C_h) \geq h$. ■

From Lemmas 8 and 9, we obtain the theorem.

5 Lower Bounds for 2-CNOT Circuits

In this section, we prove the following.

Theorem 3 *There exists an $n \times n$ 2-CNOT circuit C such that $\tau(C) = \Omega(\log n)$.* ■

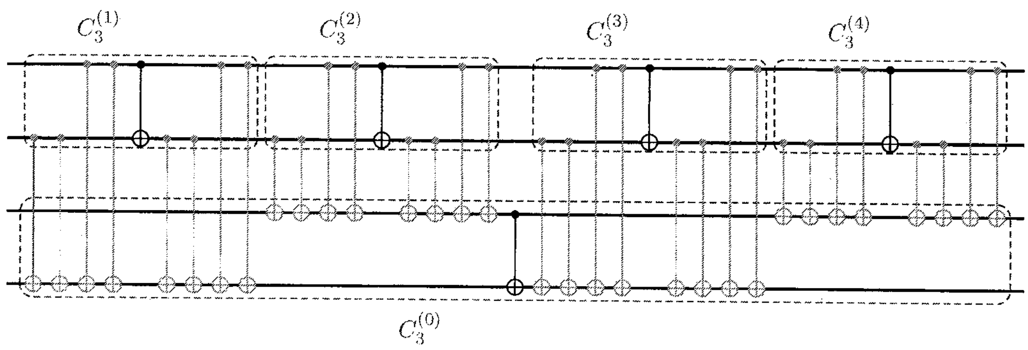


Figure 6: 4×4 1-CNOT circuit C_4.

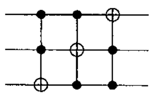


Figure 7: 3×3 2-CNOT circuit E_3.

5.1 Preliminaries

It is easy to see the following lemmas.

Lemma 10 *If E_3 is a 3×3 2-CNOT circuit shown in Fig. 7, then $\tau(E_3) = 3$.* ■

Lemma 11 *If F is a 3×3 2-CNOT circuit shown in Fig. 8, then $F(\boldsymbol{v}) = \boldsymbol{v}$ for any $\boldsymbol{v} \in \{0,1\}^3$.* ■

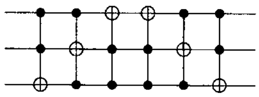


Figure 8: 3×3 2-CNOT circuit F.

5.2 Proof of Theorem 3

We prove the theorem by constructing such circuit. Let E_h ($h \geq 3$) be a 2-CNOT circuit defined as follows. Let E_3 be a 2-CNOT circuit shown in Fig. 7. For $h \geq 4$, E_h is recursively defined as follows. Let $E_{h-1}^{(i)}$ for $0 \leq i \leq \varpi_{h-1}$ and $E_{h-1}^{(j,k)}$ for $j,k \in [\varpi_{h-1}]$ be a copy of E_{h-1}, where $\varpi_{h-1} = |W(E_{h-1})|$. Construct $n_{h-1} \times n_{h-1}$ 2-CNOT circuits H_{h-1} by concatenating $E_{h-1}^{(1)}, E_{h-1}^{(2)}, \ldots, E_{h-1}^{(\varpi_{h-1})}$ and J_{h-1} by concatenating $E_{h-1}^{(1,1)}, E_{h-1}^{(1,2)} \ldots, E_{h-1}^{(1,\varpi_{h-1})}$, $E_{h-1}^{(2,1)} E_{h-1}^{(2,2)} \ldots, E_{h-1}^{(2,\varpi_{h-1})}, \ldots, E_{h-1}^{(\varpi_{h-1},\varpi_{h-1})}$, where n_{h-1} is the number of input wires of E_{h-1}.

Let $W(E_{h-1}^{(i)}) = \{w_1^{(i)}, w_2^{(i)}, \ldots, w_{\varpi_{h-1}}^{(i)}\}$ and $W(E_{h-1}^{(j,k)}) = \{w_1^{(j,k)}, w_2^{(j,k)}, \ldots, w_{\varpi_{h-1}}^{(j,k)}\}$ such that if the level of $w_i^{(*)}$ is not greater than the level of $w_j^{(*)}$, then $i \leq j$. E_h is constructed from J_{h-1}, H_{h-1}, and $E_{h-1}^{(0)}$ by inserting a copy of F for each wire $w_k^{(i,j)}$ with $i,j,k \in [\varpi_{h-1}]$ such that $w_k^{(i,j)}$ of $E_{h-1}^{(i,j)}$ in J_{h-1} is the top bit of the copy of F, $w_j^{(i)}$ of $E_{h-1}^{(i)}$ in H_{h-1} is the middle bit of the copy of F, and $w_i^{(0)}$ of $E_{h-1}^{(0)}$ is the bottom bit of the copy of F.

From the definition of E_h, we have

$$n_h = 3^{h-3} \tag{9}$$

Thus we obtain the following.

Lemma 12 $h = \Omega(\log n_h)$. ■

Lemma 13 $\tau(E_h) \geq h$.

Proof. The proof is by induction on h. $\tau(E_3) = 3$ by Lemma 10. Suppose $\tau(E_{h-1}) \geq h - 1$. We will show that $\tau(E_h) \geq h$. Suppose contrary that $\tau(E_h) = h - 1$, and let $T = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{h-1}\}$ be a complete test set for E_h. Since $\tau(E_{h-1}) \geq h - 1$, $W(E_{h-1})$ is not controllable by $T' = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{h-2}\}$. Thus there exist $i,j,k \in [\varpi_{h-1}]$ such that none of $w_i^{(0)}$, $w_j^{(i)}$, and $w_k^{(i,j)}$ is controllable by T'. It follows that if G is a copy of F with the top bit on $w_k^{(i,j)}$, then $W(G)$ is not controllable by T, a contradiction. Thus, we have $\tau(E_h) \geq h$. ■

From Lemmas 12 and 13, we obtain the theorem.

6 Concluding Remarks

It should be noted that (3) is merely an existential upper bound. It is an interesting open problem to find a polynomial time algorithm to construct a complete test of such size.

References

[1] C. Bennett, "Logical reversiblity fo computation," IBM J. Res. Dev., pp.525–532, 1973.

[2] A. Chakraborty, "Synthesis of reversible circuits for testing with universal test set and c-testability of reversible iterative logic array," Proc. of the 18h International Conference on VLSI Design, 2005.

[3] R. Merkle, "Two types of mechanical reversible logic," Nanotechnology, pp.114–131, 1993.

[4] M. Nielsen and I. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, 2000.

[5] K. Patel, J. Hayes, and I. Markov, "Fault testing for reversible circuits," IEEE Trans. Computer-Aided Design, pp.1220–1230, Aug. 2004.

[6] V. Shende, A. Prasad, I. Markov, and J. Hayes, "Synthesis of reversible logic circuits," IEEE Trans. Computer-Aided Design, pp.710–722, june 2003.