Fault Testing for Reversible Circuits

Ketan Patel

November 6, 2002

Outline

- Motivation
- Reversible Circuits
- Fault Models
- Test Set Constructions (Upper Bounds)
- Integer Linear Programming Formulation
- Future Work

Motivation

Reversible Circuits

- “Energy-free” computation
- Reversible applications: cryptography, DSP, etc.
- Important class of quantum circuits

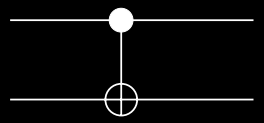
Efficient testing will be important

Reversible Gates (Examples)

NOT Gate

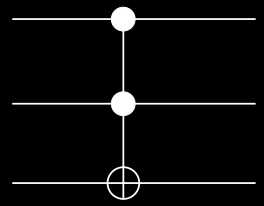
Input	Output
0	1
1	0

C–NOT Gate



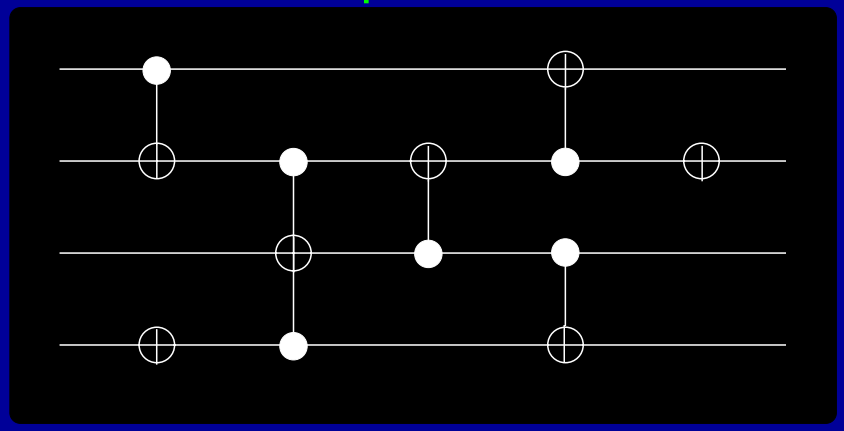
Input	Output
00	00
01	01
10	11
11	10

TOFFOLI Gate



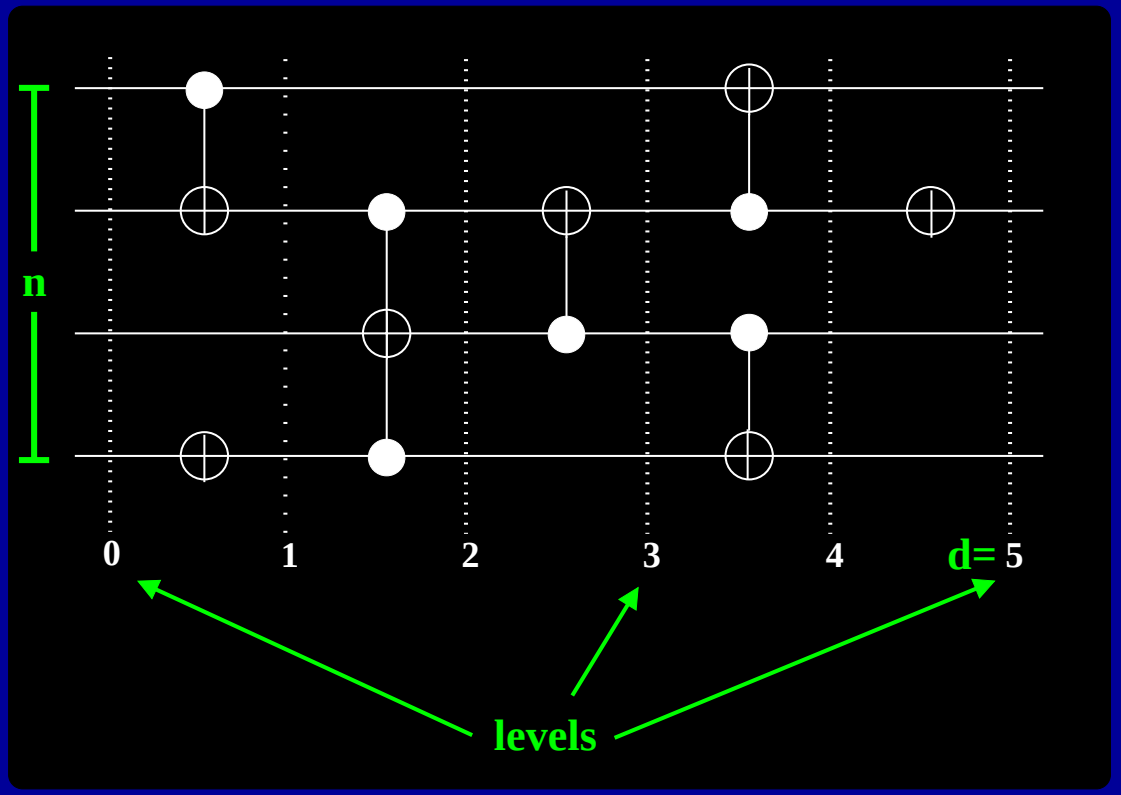
Input	Output
000	000
001	001
010	010
011	011
100	100
101	101
110	111
111	110

Sample Circuit



Notation & Terminology

n	# of inputs/outputs
l	# of gates
k_i	size of ith gate
d	circuit depth
f_j	function of first j levels
f_j^{-1}	inverse function



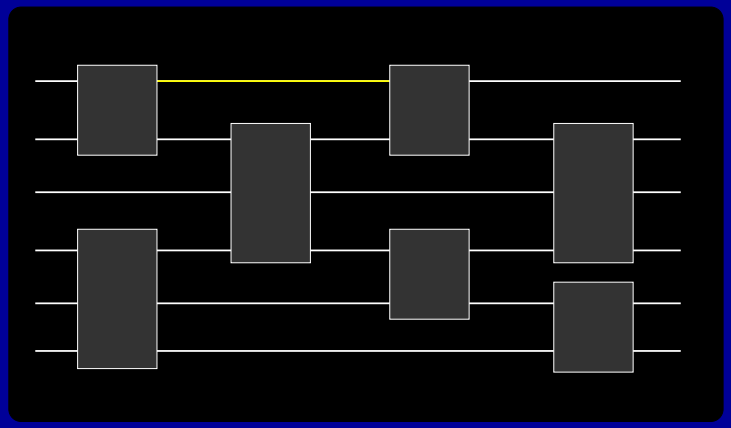
Properties of Reversible Circuits

Controllability: Wires at any level can be set to any state

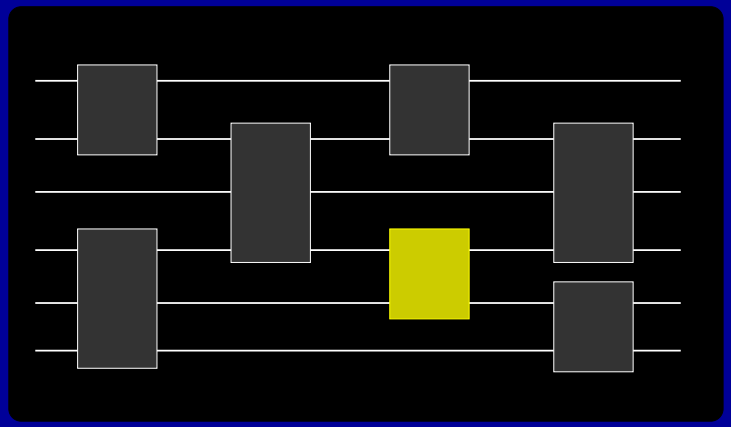
Observability: Any change in intermediate state changes output

Fault Models

Stuck-at Model: Wire stuck at 0 (or 1)



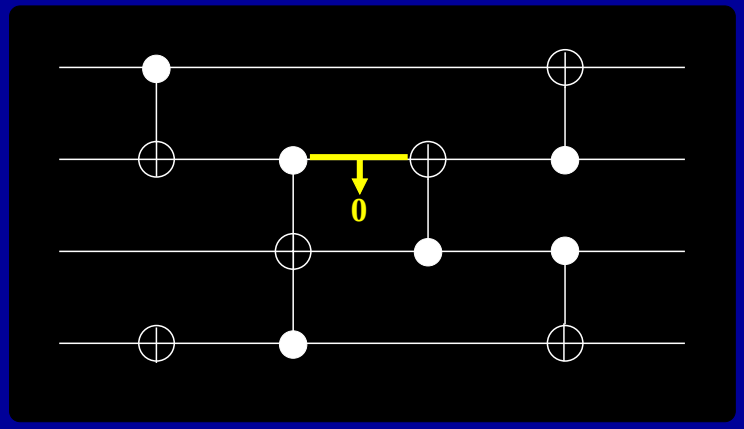
Cell Fault Model: Single component fails



Goal

Generate (minimal) set of input vectors which can detect any fault in F

Example:

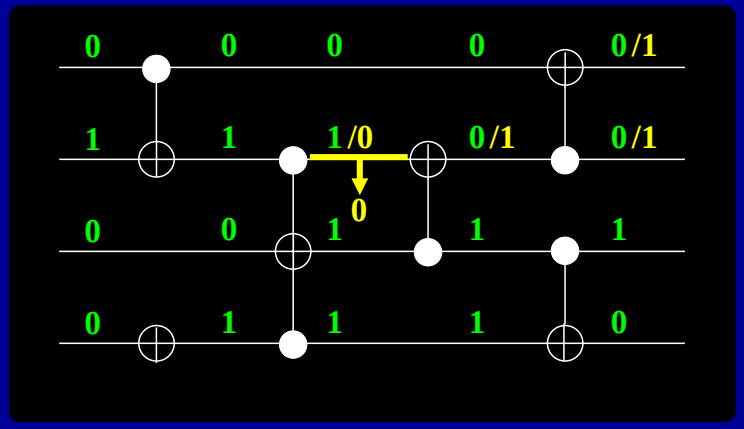


Use vector 0100 to detect fault

Goal

Generate (minimal) set of input vectors which can detect any fault in F

Example:



Use vector 0100 to detect fault

Correct output: 0010

Faulty output: 1110

Test set is complete if it detects all faults in set

General Conditions

Stuck-at Fault Model:

Test set is complete $\Leftrightarrow$ every fault site can be set to both 0 and 1

$$\text{\# of faults } = 2 \cdot \left(n + \sum_{i=1}^{l} k_i \right)$$

Cell Fault Model:

Test set is complete $\Leftrightarrow$ all input patterns occur at input of every gate

$$\text{\# of faults } = \sum_{i=1}^{l} 2^{k_i}$$

Here we focus on stuck-at model

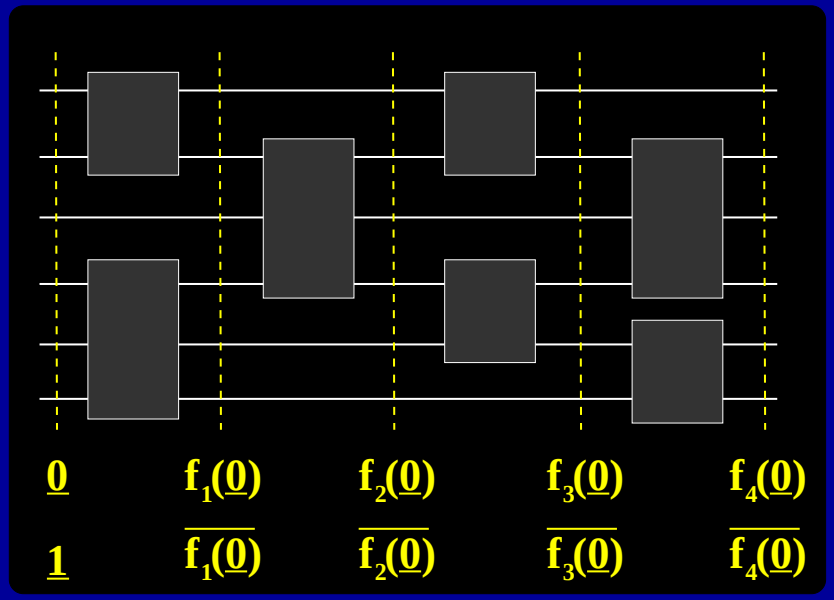
Basic Upper Bounds

$|T| \leq 2^{n-1} + 1$

Each fault site set to 0 (or 1) by half input vectors

$|T| \leq d + 1$

$\left\{\underline{0},\ \underline{1},\ f_1^{-1}(\overline{f_1(\underline{0})}), \ldots,\ f_d^{-1}(\overline{f_d(\underline{0})})\right\}$



Efficient Test Sets Exist

Lemma If test set covers C faults
$\Rightarrow \exists$ input vector covering $\geq \frac{1}{2}$ remaining faults.

Corollary

$$|T| \leq \left\lfloor \log_2 \left(n + \sum_{i=1}^{l} k_i \right) \right\rfloor + 2$$

$\Rightarrow$ Efficient test set exists

Example: if $n = 64$, $l = 10^6$, and $k_i = 3$ then $|T| \leq 23$

...but proof "non-constructive"

Integer Linear Programming Formulation

t_i	decision variable representing input vector i
T_i	n-bit expansion of int i
$f_j(T_i)$	values at level j for input T_i

Minimize $t_0 + t_1 + \cdots + t_{2^n-1}$

subject to the constraints

$$\sum_i f_j(T_i) \cdot t_i \geq 1$$

$$\sum_i \overline{f_j(T_i)} \cdot t_i \geq 1, \quad 0 \leq j \leq d$$

where $t_i \in \{0,1\}$, $0 \leq i \leq 2^n - 1$

2^n variables and $2n(d+1)$ constraints $\Rightarrow$ infeasible for large circuits

C-NOT Example

Minimize $t_0 + t_1 + t_2 + t_3$

subject to the constraints

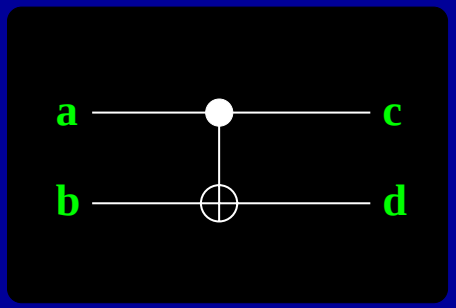
$a)$ $t_0 + t_1 \geq 1 \quad t_2 + t_3 \geq 1$

$b)$ $t_0 + t_2 \geq 1 \quad t_1 + t_3 \geq 1$

$c)$ $t_0 + t_1 \geq 1 \quad t_2 + t_3 \geq 1$

$d)$ $t_0 + t_3 \geq 1 \quad t_1 + t_2 \geq 1$

where $t_0,\ t_1,\ t_2,\ t_3 \in \{0,1\}$

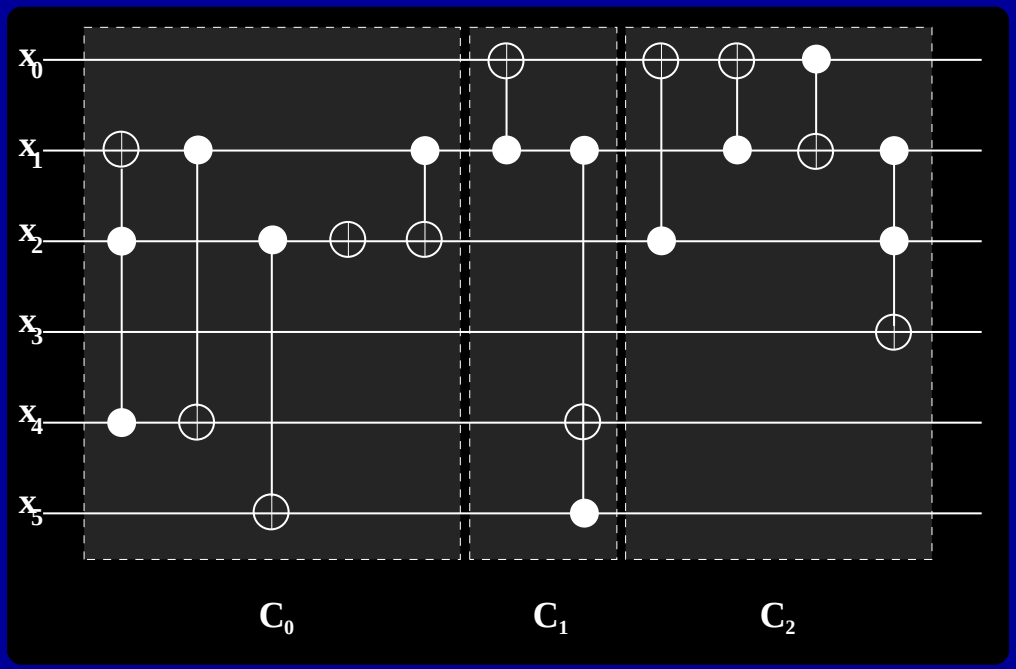


$\Rightarrow$ need 3 test vectors

Circuit Decomposition

Alternate approach:

- decompose circuit into shorter subcircuits acting on fewer wires
- iteratively apply ILP method
- dynamically combine test sets



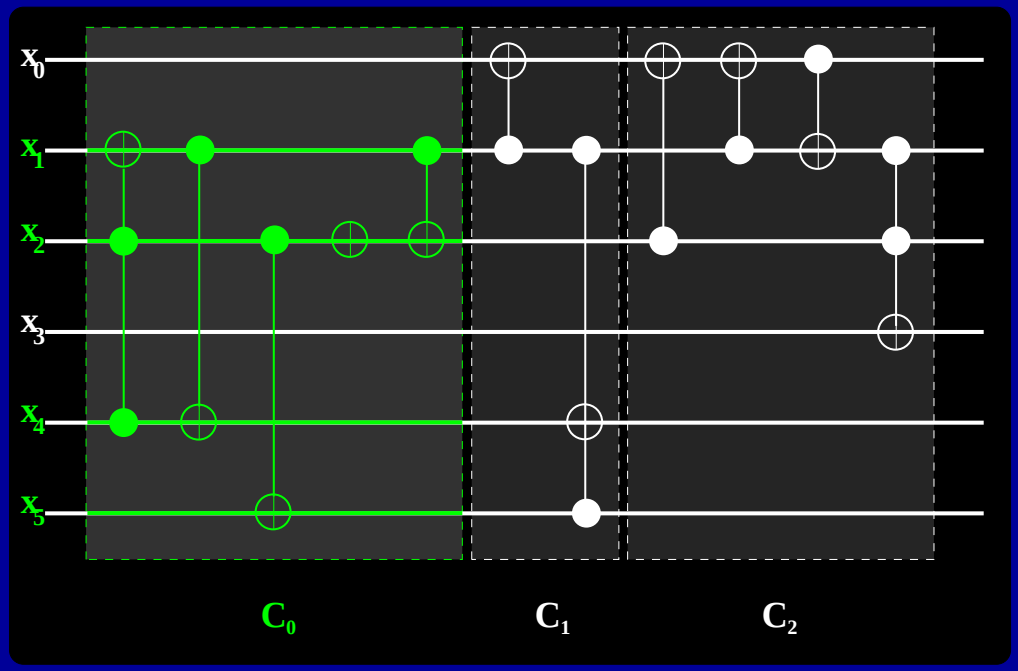
Circuit Decomposition

Solving ILP for C_0 gives

x_0	x_1	x_2	x_3	x_4	x_5
X	0	1	X	1	1
X	1	0	X	0	0
X	1	1	X	1	0

Passing these through C_0 gives

x_0	x_1	x_2	x_3	x_4	x_5
X	1	1	X	0	0
X	1	0	X	1	0
X	0	0	X	1	1



Circuit Decomposition

Generate ILP for C_1 & add constraints

x_0	x_1	x_4	x_5
X	1	0	0
X	1	1	0
X	0	1	1

$\in T \Rightarrow$

Constraints
$t_4 + t_{12} \geq 1$
$t_6 + t_{14} \geq 1$
$t_3 + t_{11} \geq 1$

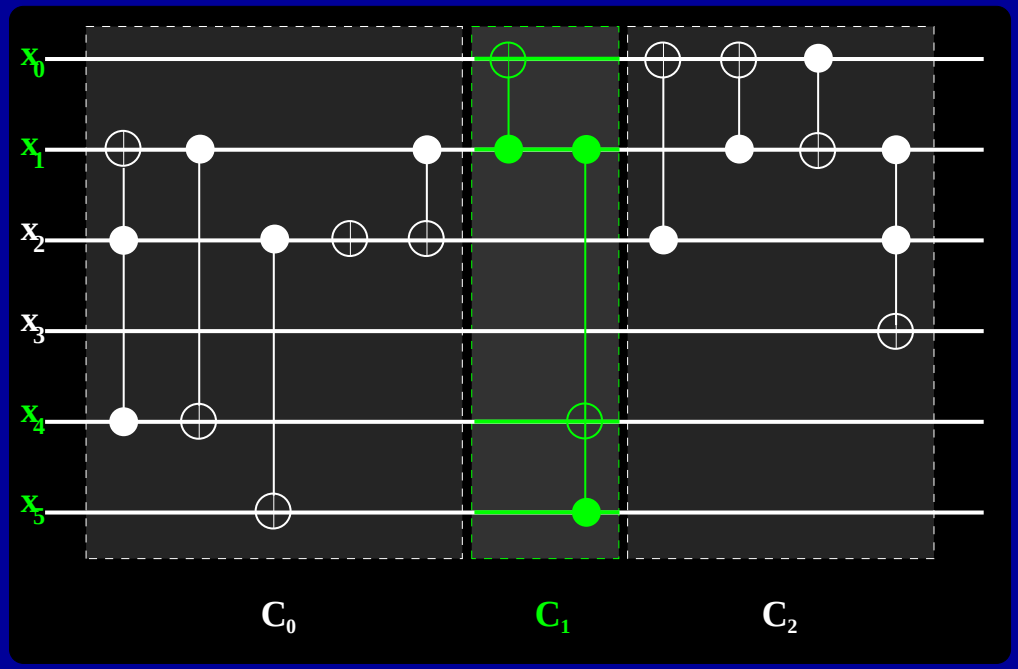
Yields solution $t_6 = t_{11} = t_{12} = 1$

New test set

x_0	x_1	x_2	x_3	x_4	x_5
1	1	1	X	0	0
0	1	0	X	1	0
1	0	0	X	1	1

Passing through C_1 gives

x_0	x_1	x_2	x_3	x_4	x_5
0	1	1	X	0	0
1	1	0	X	1	0
1	0	0	X	1	1



Circuit Decomposition

Generate ILP for C_2 & add constraints

x_0	x_1	x_4	x_5
0	1	1	X
1	1	0	X
1	0	0	X

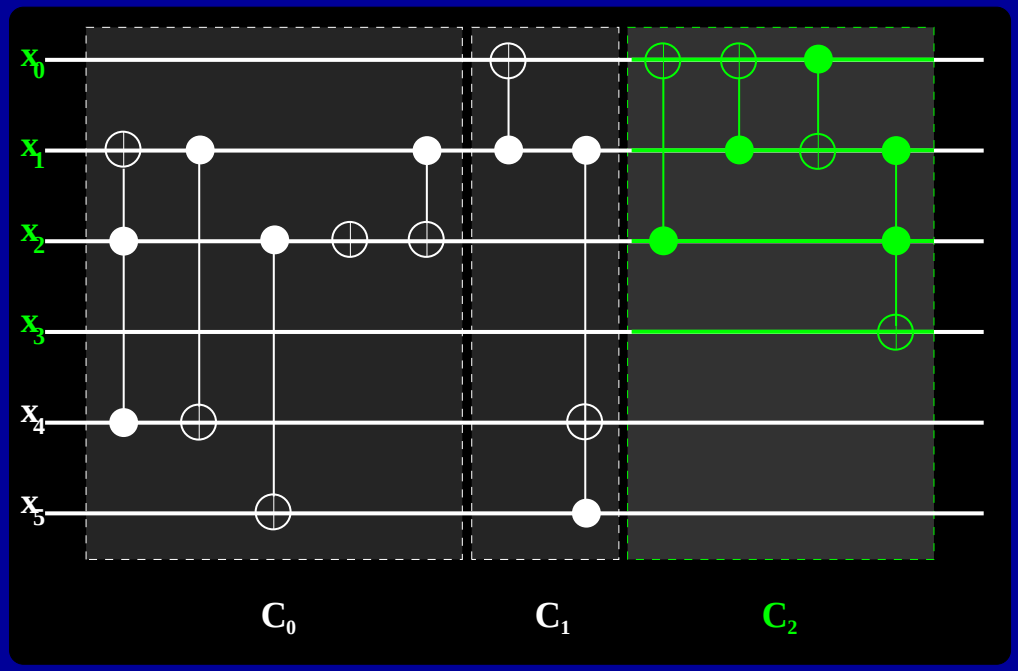
$\in T \Rightarrow$

Constraints		
t_6	$+ t_7$	≥ 1
t_{12}	$+ t_{13}$	≥ 1
t_8	$+ t_9$	≥ 1

Yields solutions $t_5 = t_7 = t_8 = t_{12} = 1$

New test set

x_0	x_1	x_2	x_3	x_4	x_5
0	1	1	1	0	0
1	1	0	0	1	0
1	0	0	0	1	1
0	1	0	1	X	X



Test Set Compaction

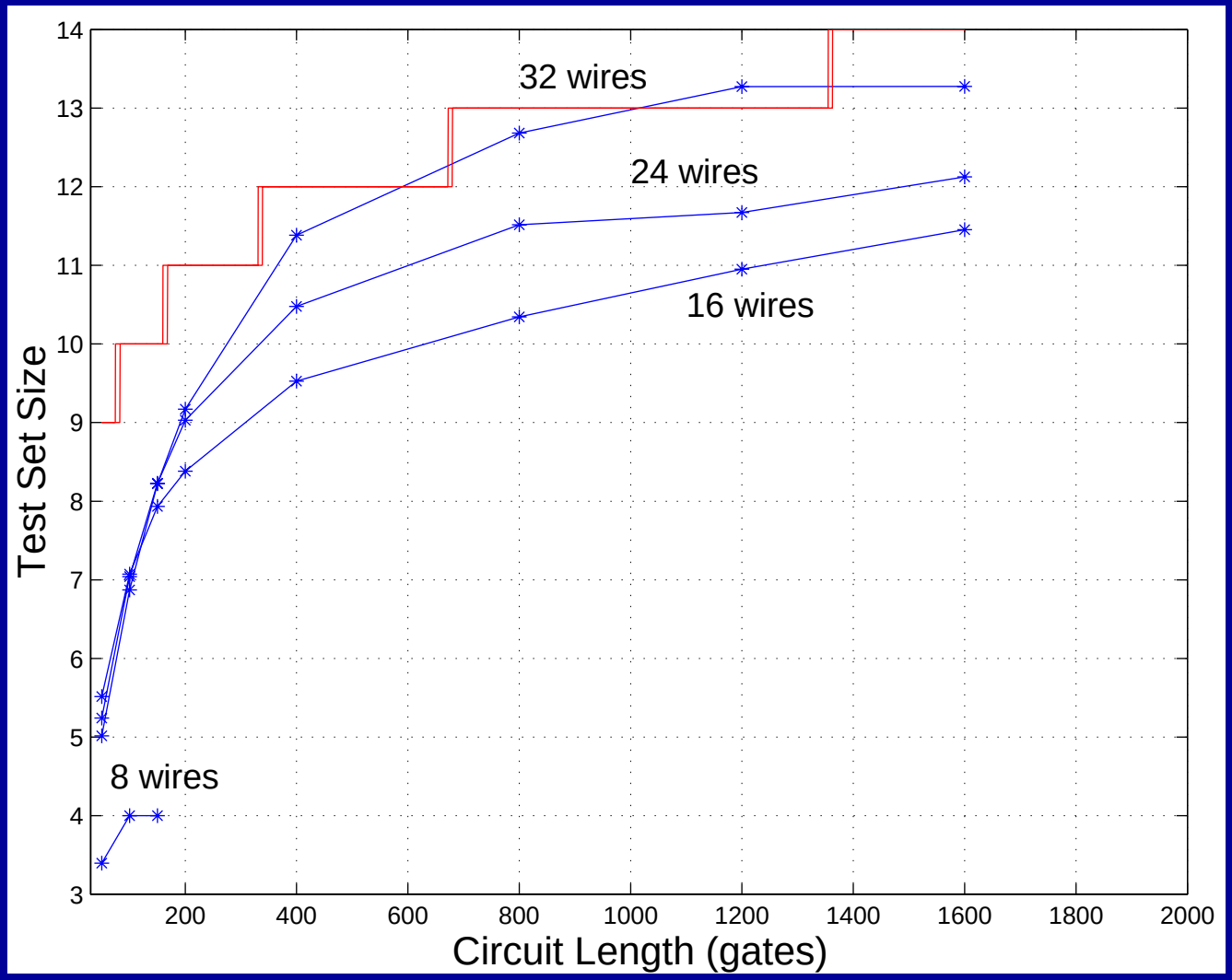
Test sets via decomposition method may be redundant

Find smallest subset of complete vectors: compaction

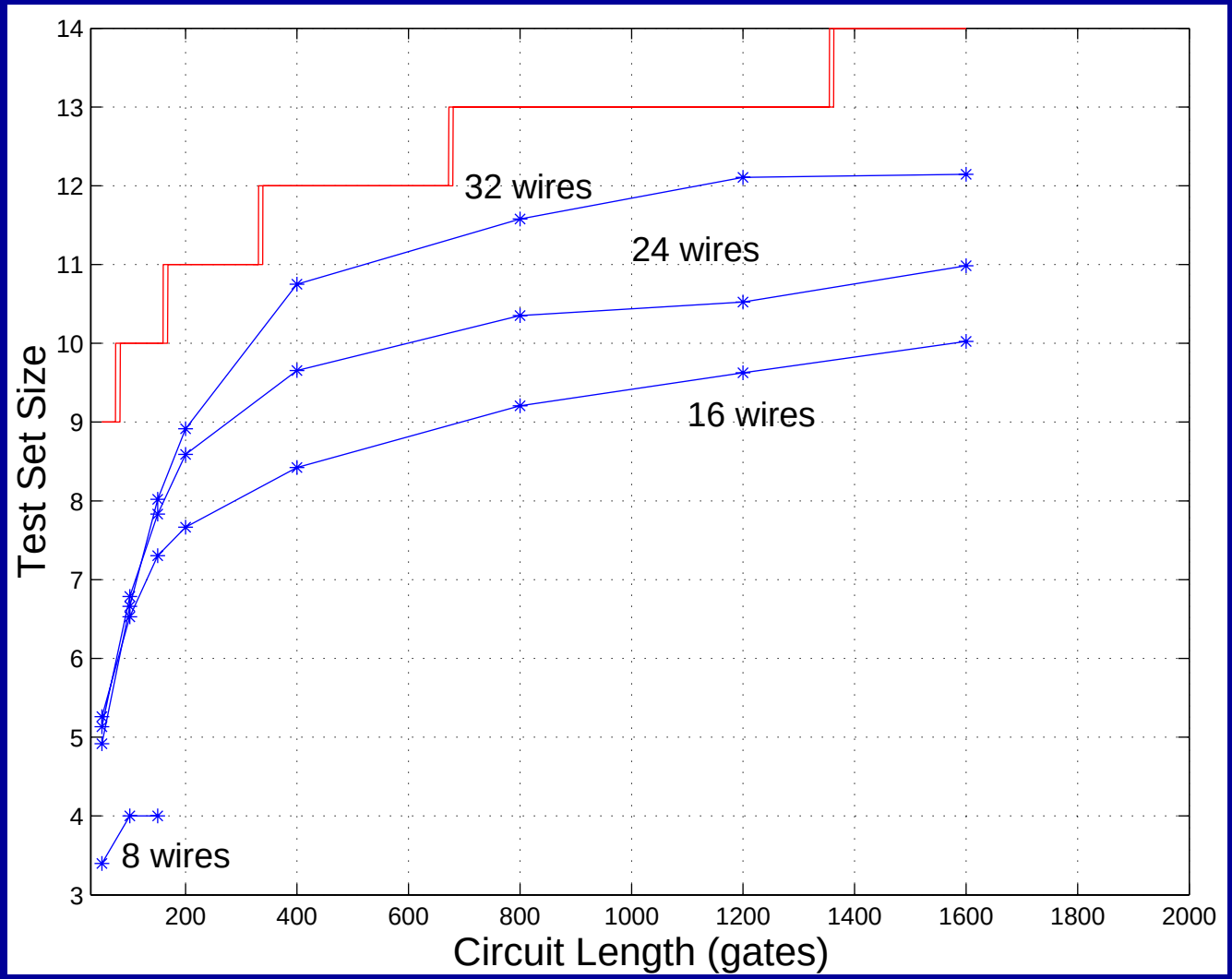
Use ILP method with subset of t_i's

Much easier to solve than overall ILP if original test set small

Simulation Results



Simulation Results (Compaction)



Cell Fault Model

$$|T| \leq 2^n - 2^{n-k_1} + 1$$

$$|T| \leq \left(\Sigma_{i=1}^{l} 2^{k_i}\right) - l + 1$$

$$|T| \leq \Sigma_{i=1}^{l} \left\lceil \frac{2^{k_i}}{i} \right\rceil$$

where circuit has l gates w/ sizes $k_1 \geq k_2 \geq \cdots k_l$

ILP and decomposition methods work as in stuck-at case

Future Work

- Lower Bounds
- Circuits with Known Minimal Test Sets
- Connections with Circuit Synthesis
- Fault Diagnosis
- Quantum Circuits