Bounds on Reliable Boolean Function Computation with Noisy Gates

- R. L. Dobrushin & S. I. Ortyukov, 1977
- N. Pippenger, 1985
- P. Gács & A. Gál, 1994

Presenter: Da Wang
6.454 Graduate Seminar in Area I
EECS, MIT

Oct. 5, 2011

Question

Given a network of noisy logic gates, what is the redundancy required if we want to compute the a Boolean function reliably?

- **noisy:** gates produce the wrong output independently with error probability no more than ε.
- **reliably:** the value computed by the entire circuit is correct with probability at least $1-\delta$
- **redundancy:**

$$\frac{\text{minimum \#gates needed for reliable computation in noisy circuit}}{\text{minimum \#gates needed for reliable computation in noiseless circuit}}$$

  - noisy/noiseless complexity
  - may depend on the function of interest
  - upper bound: achievability
  - lower bound: converse

Part I

Lower Bounds for the Complexity of Reliable Boolean Circuits with Noisy Gates

History of development

- [Dobrushin & Ortyukov 1977]
  - Contains all the key ideas
  - Proofs for a few lemmas are incorrect
- [Pippenger & Stamoulis & Tsitsiklis 1990]
  - Pointed out the errors in [DO1977]
  - Provide proofs for the case of computing the parity function
- [Gács & Gál 1994]
  - Follow the ideas in [DO1977] and provide correct proofs
  - Also prove some stronger results

In this talk
We will mainly follow the presentation in [Gács & Gál 1994].

Problem formulation
System Model

Boolean circuit C

- a directed acyclic graph
- node $\sim$ gate
- edge $\sim$ in/out of a gate

Gate g

- a function $g : \{0,1\}^{n_g} \to \{0,1\}$
  - n_g: fan-in of the gate

Basis Φ

- a set of possible gate functions
- e.g., $\Phi = \{AND, OR, XOR\}$
- complete basis
- for circuit C: Φ_C
- maximum fan-in in C: $n(\Phi_C)$

Assumptions

- each gate g has constant number of fan-ins n_g.
- f can be represented by compositions of gate functions in Φ_C.

Problem formulation

Error models (ε, p)

Gate error

- A gate **fails** if its output value for $\mathbf{z} \in \{0,1\}^{n_g}$ is different from $g(\mathbf{z})$
- gates fail independently with
  - fixed probability ε
    - used for lower bound proof
  - probability at most ε
- $\varepsilon \in (0, 1/2)$

Circuit error

- $C(\mathbf{x})$: random variable for output of circuit C on input $\mathbf{x}$.
- A circuit computes f with error probability at most p if

$$\mathbb{P}\left[C(\mathbf{x}) \neq f(\mathbf{x})\right] \leq p$$

for any input $\mathbf{x}$.

Problem formulation

Sensitivity of a Boolean function

Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function with binary input vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)$.
Let $\mathbf{x}^l$ be a binary vector that differs from $\mathbf{x}$ only in the l-th bit, i.e.,

$$\mathbf{x}_i^l = \begin{cases} x_i & i \neq l \\ \neg x_i & i = l \end{cases}.$$

- f is sensitive to the lth bit on $\mathbf{x}$ if $f(\mathbf{x}^l) \neq f(\mathbf{x})$.
- Sensitivity of f on $\mathbf{x}$: #bits in $\mathbf{x}$ that f is sensitive to.
  - “effective” input size
- Sensitivity of f: maximum over all $\mathbf{x}$.

Asymptotic notations

- $f(n) = O(g(n))$:

$$\limsup_{n\to\infty} \left| \frac{f(n)}{g(n)} \right| < \infty,$$

- $f(n) = \Omega(g(n))$:

$$\liminf_{n\to\infty} \left| \frac{f(n)}{g(n)} \right| \geq 1,$$

- $f(n) = \Theta(g(n))$:

$$f(n) = O(g(n))$$

and

$$f(n) = \Omega(g(n))$$

Main results

Theorem: number of gates for reliable computation

- Let ε and p be any constants such that $\varepsilon \in (0, 1/2), p \in (0, 1/2)$.
- Let f be any Boolean function with sensitivity s.

Under the error model (ε, p), the number of gates of the circuit is $\Omega(s \log s)$.

Corollary: redundancy of noisy computation

For any Boolean function of n variables and with $O(n)$ noiseless complexity and $\Omega(n)$ sensitivity, the redundancy of noisy computation is $\Omega(\log n)$.

- e.g., nonconstant symmetric function of n variables has redundancy $\Omega(\log n)$

Equivalence result for wire failures

Lemma 3.1 in Dobrushin&Ortyukov

- Let $\varepsilon \in (0, 1/2)$ and $\delta \in [0, \varepsilon/n(\Phi_C)]$.
- Let $\mathbf{y}$ and $\mathbf{t}$ be the vector that a gate receives when the wire fail and does not fail respectively.

For any gate g in the circuit C there exists unique values $\eta_g(\mathbf{y}, \delta)$ such that if

- the wires of C fails independently with error probability δ, and
- the gate g fails with probability $\eta_g(\mathbf{y}, \delta)$ when receiving input $\mathbf{y}$,

then the probability that the output of g is different from $g(\mathbf{t})$ is equal to ε.

Insights

- Independent gate failures can be "simulated" by independently wire failures and corresponding gate failures.
- These two failure modes are equivalent in the sense that the circuit C computes f with the same error probability.

“Noisy-wires” version of the main result

Theorem

- Let ε and p be any constants such that $\varepsilon \in (0, 1/2), p \in (0, 1/2)$.
- Let f be any Boolean function with sensitivity s.

Let C be a circuit such that

- its wires fail independently with fixed probability δ, and
- each gate fails independently with probability $\eta_g(\mathbf{y}, \delta)$ when receiving $\mathbf{y}$.

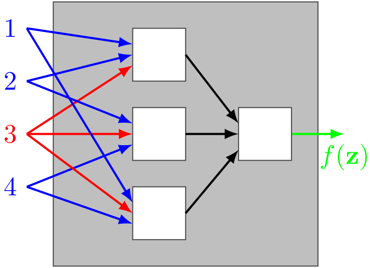
Suppose C computes f with error probability at most p. Then the number of gates of the circuit is $\Omega\left(s \log s\right)$.

Maximal sensitive set S for f

- $s > 0$: sensitivity of f
- $\mathbf{z}$: an input vector with s bits that f is sensitive to
  - an input vector where f has maximum sensitivity
- S: the set of sensitive bits in $\mathbf{z}$
  - key object

B_l: edges originated from l-th input

- $m_l \triangleq |B_l|$
- e.g.
  - $l = 3$
  - B_l
  - $m_l = 3$



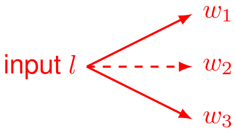
Error analysis

Wire failures

- For $\beta \subset B_l$, let $H(\beta)$ be the event that for wires in B_l, only those in β fail.
- Let

$$\beta_l \triangleq \arg\max_{\beta \subset B_l} \mathbb{P}\left[C(\mathbf{z}^l) = f(\mathbf{z}^l) \,\middle|\, H(\beta)\right]$$

  - the best failing set for input $\mathbf{z}^l$
- Let $H_l \triangleq H(B_l \setminus \beta_l)$



- $B_l = \{w_1, w_2, w_3\}$
- $\beta = \{w_2\}$

Fact 1

$$\mathbb{P}\left[C(\mathbf{z}) \neq f(\mathbf{z}) \mid H_l\right] = \mathbb{P}\left[C(\mathbf{z}^l) = f(\mathbf{z}^l) \,\middle|\, H(\beta_l)\right]$$

- Proof
  - f is sensitive to z_l
  - $\neg z_l \Leftrightarrow$ “flip” all wires in B_l
- β_l is the worst non-failing set for input $\mathbf{z}$

Fact 2

$$\mathbb{P}\left[C(\mathbf{z}^l) = f(\mathbf{z}^l) \,\middle|\, H(\beta_l)\right] \geq 1 - p$$

- Proof
  - $\mathbb{P}\left[C(\mathbf{z}^l) = f(\mathbf{z}^l)\right] \geq 1 - p$
  - β_l maximizes $\mathbb{P}\left[C(\mathbf{z}^l) = f(\mathbf{z}^l) \,\middle|\, H(\beta)\right]$

Fact 1 & 2 $\Rightarrow$ Fact 3

For each $l \in S$,

$$\mathbb{P}\left[C(\mathbf{z}) \neq f(\mathbf{z}) \,|\, H_l\right] \geq 1 - p$$

where $\{H_l, l \in S\}$ are independent events. Furthermore, Lemma 4.3 in [Gács&Gál 1994] shows

$$\mathbb{P}\left[C(\mathbf{z}) \neq f(\mathbf{z}) \,\middle|\, \bigcup_{l \in S} H_l\right] \geq (1 - \sqrt{p})^2$$

- The error probability given H_l or $\bigcup_{l \in S} H_l$ is relatively large.

Note

$$
\begin{aligned}
p &\geq \mathbb{P}\left[C(\mathbf{z}) \neq f(\mathbf{z})\right] \\
&\geq \mathbb{P}\left[C(\mathbf{z}) \neq f(\mathbf{z}) \,\middle|\, \bigcup_{l \in S} H_l\right] \mathbb{P}\left[\bigcup_{l \in S} H_l\right]
\end{aligned}
$$

Fact 3 implies

Fact 4

$$
\mathbb{P}\left[\bigcup_{l \in S} H_l\right] \leq \frac{p}{(1-\sqrt{p})^2}
$$

which implies (via Lemma 4.1 in [Gács&Gál 1994]),

Fact 5

$$
\mathbb{P}\left[\bigcup_{l \in S} H_l\right] \geq \left(1 - \frac{p}{(1-\sqrt{p})^2}\right) \sum_{l \in S} \mathbb{P}\left[H_l\right]
$$

Error analysis

Bounds on the total number of sensitive wires

Fact 6

$$\mathbb{P}\left[H_l\right] = (1-\delta)^{|\beta_l|}\delta^{m_l - |\beta_l|} \geq \delta^{m_l}$$

Fact 4 & 5 $\Rightarrow$

$$\begin{aligned} \frac{p}{1-2\sqrt{p}} &\geq \sum_{l \in S} \delta^{m_l} \\ &\geq s\left(\prod_{l \in S} \delta^{m_l}\right)^{1/s} \end{aligned}$$

which leads to

$$\sum_{l \in S} m_l \geq \frac{s}{\log(1/\delta)} \log\left(s\frac{1-2\sqrt{p}}{p}\right)$$

- lower bound on the total number of “sensitive wires”

Lower bound on number of gates

Let N_C be the total number of gates in C:

$$n(\Phi_C)N_C \geq \sum_g n_g \geq \sum_{l \in S} m_l \geq \frac{s}{\log(1/\delta)} \log\left(s\frac{1-2\sqrt{p}}{p}\right)$$

Comments:

- The above proof is for $p \in (0, 1/4)$
- The case $p \in (1/4, 1/2)$ can be shown similarly.

Block Sensitivity

Let $\mathbf{x}^S$ be a binary vector that differs from $\mathbf{x}$ in the S subset of indices, i.e.,

$$\mathbf{x}_i^S = \begin{cases} x_i & i \notin S \\ \neg x_i & i \in S \end{cases}.$$

- f is (block) sensitive to S on $\mathbf{x}$ if $f(\mathbf{x}^S) \neq f(\mathbf{x})$.
- Block sensitivity of f on $\mathbf{x}$: the largest number b such that
  - there exists b disjoint sets $S_1, S_2, \cdots, S_b$
  - for all $1 \leq i \leq b$, f is sensitive to S_i on $\mathbf{x}$
- Block sensitivity of f: maximum over all $\mathbf{x}$.
  - block sensitivity $\geq$ sensitivity

Theorem based on block sensitivity

- Let ε and p be any constants such that $\varepsilon \in (0, 1/2), p \in (0, 1/2)$.
- Let f be any Boolean function with block sensitivity b.

Under the error model (ε, p), the number of gates of the circuit is $\Omega\left(b \log b\right)$.

Given an explicit function f of n variables, is there a lower bound that is stronger than $\Omega\left(n \log n\right)$?

Open problem for

- unrestricted circuit C with complete basis
- function f that have $\Omega\left(n \log n\right)$ noiseless complexity for circuit C with some incomplete basis Φ

Exponential blowup

A noisy circuit with multiple levels

- The output of gates at level l goes to a gate at level $l+1$
- Level 0 has n inputs
  - Level 0 has $N_0 = n \log n$ output gates
  - Level 1 has N_0 inputs
  - Level 1 has $N_1 = N_0 \log N_0$ output gates, ...

Why?

"The theorem is generally applicable only to the very first step of such a fault tolerant computation"

- If the input is not the original ones, we can choose them to make the sensitivity of a Boolean function to be 0.
  - $f(x_1, x_2, x_3, x_4, x_1 \oplus x_2 \oplus x_4, x_1 \oplus x_3 \oplus x_4, x_2 \oplus x_3 \oplus x_4)$
  - Lower bound does not apply: sensitivity is 0. How about block sensitivity?
- Problem formulation issue on the lower bound for coded input
  - coding is also computation!

Part II

Upper Bounds for the Complexity of Reliable Boolean Circuits with Noisy Gates

[Pippenger, “On Networks of Noisy Gates”, 1985]

Overview

Achievability schemes in reliable computation with a network of noisy gates.

1. System modeling
   - various types of computations
2. Change of basis and error levels
   - will skip
3. Functions with logarithmic redundancy
   - with explicit construction
   - for specific system parameters only
4. Functions with bounded redundancy
   - Presents a class of functions with “bounded redundancy”
   - Construction for reliable computation

perturbation and approximation

Let $f, g : \{0,1\}^k \Rightarrow \{0,1\}$,

- g is a ε-perturbation of f if $\mathbb{P}\left[g(\mathbf{x}) = f(\mathbf{x})\right] = 1 - \varepsilon$ for any $\mathbf{x} \in \{0,1\}^k$
- g is a ε-approximation of f if $\mathbb{P}\left[g(\mathbf{x}) = f(\mathbf{x})\right] \geq 1 - \varepsilon$ for any $\mathbf{x} \in \{0,1\}^k$

weakly (ε, δ)-computes

- gates: ε-perturbation
- output: δ-approximation

strongly (ε, δ)-computes

- gates: ε-approximation
- output: δ-approximation

Why bother?

- ε-perturbation may be helpful in randomized algorithms.

Theorem 3.1

If a Boolean function is computed by a noiseless network of size c, then it is also computed by a noisy network of size $O\left(c \log c\right)$.

Comments

- Provides explicit construction for some ε and δ values.
    - $\varepsilon = 1/512$
    - $\delta = 1/128$

Strategy

Given a noiseless network with 2-input gates, construct a corresponding noisy network with 3-input gates.

Transformations

noiseless		noisy
each wire	$\rightarrow$	cable of m wires
gate	$\rightarrow$	module of $O(m)$ noisy gates

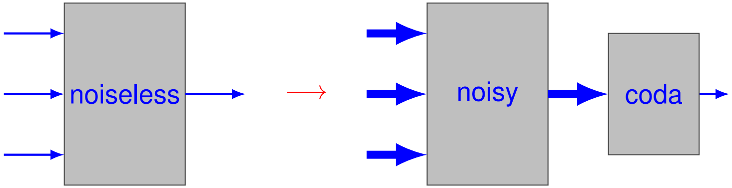
Additions

- coda: computes the majority of m wires with at most some error probability
  - Corollary 2.6: exists coda with size $O\left(c \log c\right)$

- Choose $m = O(\log c)$

- a cable is correct if at least $(1-\theta)m$ component wires are correct

Overview



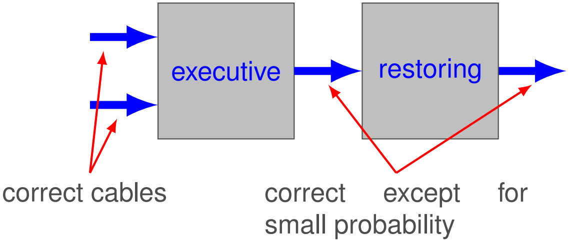
Module requirement

If the input cables are “correct”, then the output cable will be correct except for some small error probability.

Idea:

- Use “modular redundancy” and majority voting
- Binomial $(1, 1-\varepsilon)$ vs. $\frac{1}{m}$Binomial $(m, 1-\varepsilon)$

Module construction



Executive organ

- Construction: m noisy gates that compute the same function as the corresponding gate in noiseless network

Restoring organ

- Construction: a (m, k, α, β)-compressor
  - if at most αm inputs are incorrect, then at most βm outputs will be incorrect.
- $k = 8^{17}$, $\alpha = 1/64$, $\beta = 1/512$

Then

Choose system parameters properly, such that the resulting circuit has logarithmic redundancy.

Functions with bounded redundancy

For $r \geq 1$, let $s = 2^r$. Let

$$g_r(x_0, \ldots, x_{r-1}, y_0, \ldots, y_{s-1}) = y_t$$

where $t = \sum_{i=0}^{r-1} 2^i x_i$ i.e., t has binary representation $x_{r-1} \cdots x_1 x_0$.

Theorem 4.1

For every r and $s = 2^r$, g_r can be computed by a network of $O(s)$ noisy gates.

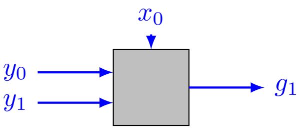
Comments

- g_r: “indicator function”
- Any noiseless networks that computes g_r has $\Omega\left(2^r\right)$ gates.
  - bounded redundancy
- Proof
  - Construct a network that strongly $(\varepsilon = 1/192, \delta = 1/24)$-computes g_r.

Construction

g_1

$$g_1(x_0, y_0, y_1) = \begin{cases} y_0 & x_0 = 0 \\ y_1 & x_1 = 1 \end{cases}$$



g_r

$$g_2(x_0, x_1, y_0, y_1, y_2, y_3) = \begin{cases} y_0 & x_1 x_0 = 00 \\ y_1 & x_1 x_0 = 01 \\ y_2 & x_1 x_0 = 10 \\ y_3 & x_1 x_0 = 11 \end{cases}$$

$$\ldots$$

- g_r can be implemented by a binary tree with $2^r - 1$ elements of g_1.
  - level $r - 2$: root
  - level 0: leaves
  - y_t: corresponds to a path from level 0 to $r - 2$

Construction (cont.)

- Each path only contains one gate at each level
- If each gate at level $k, 0 \le k \le r-2$ fails with probability $\Theta\left((a\varepsilon)^k\right)$, then the failure probability for a path is $\Theta(\varepsilon)$.

Construction: replace wires by cables, gates by modules

- cable at level k
  - input: $2k-1$ wires
  - output: $2k+1$ wires
- module at level k
  - $2k+1$ disjoint networks
  - each compute the $(2k-1)$-argument majority of the input wires
  - then apply g_1
  - noiseless complexity: $O(k) \Rightarrow$ noisy complexity: $O(k \log k)$
    - $O\left(k^2 \log k\right)$ noisy gates at level k
  - error probability for each noisy network: 2ε
    - error probability for module: $4\varepsilon(8\varepsilon)^k = \Theta\left((8\varepsilon)^k\right)$
- use coda at the root output for majority vote
- total #gate: $O(s) = O\left(2^r\right)$

A network with outputs $w_1, w_2, \ldots, w_m$ strongly (ε, δ)-computes $f_1, f_2, \ldots, f_m$ if, for every $1 \leq j \leq m$, the network obtained by ignoring all but the output w_j strongly (ε, δ)-computes f_j.

Theorem 4.2

For every $a \geq 1$ and $b = 2^{2^a}$, let $h_{a,0}(z_0, \cdots, z_{a-1}), \cdots, h_{a,b-1}(z_0, \cdots, z_{a-1})$ denote the b Boolean functions of a Boolean argument.

Then $h_{a,0}(z_0, \cdots, z_{a-1}), \cdots, h_{a,b-1}(z_0, \cdots, z_{a-1})$ can be strongly computed by a network of $O(b)$ noisy gates.

- Proof: similar to Theorem 4.1

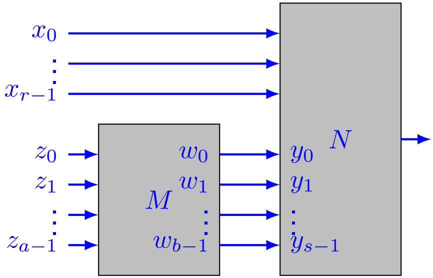
Boolean function with n Boolean arguments

Theorem 4.3

Any Boolean function of n Boolean arguments can be computed by a network of $O\left(2^n/n\right)$ noisy gates.

Proof

- Let $a = \lfloor \log_2(n - \log_2 n) \rfloor$, $b = 2^{2^a} = 2^n/n$, $r = n - a$ and $s = 2^r = 2^n/n$.
- Theorem 4.2: M strongly computes $h_{a,0}(z_0, \cdots, z_{a-1})$, $\cdots$, $h_{a,b-1}(z_0, \cdots, z_{a-1})$
  - $O(b) = O\left(2^n/n\right)$ gates
- Theorem 4.1: N strongly computes $g_r(x_0, \ldots, x_{r-1}, y_0, \ldots, y_{s-1})$
  - $O(s) = O\left(2^n/n\right)$ gates



M and N: strongly computes any Boolean function with n Boolean arguments $x_0, x_1, \cdots, x_{r-1}, z_0, z_1, \cdots, z_{a-1}$.

Bounded redundancy for Boolean functions

Implication of Theorem 4.3

- [Muller, "Complexity in Electronic Switching Circuits", 1956]: "Almost all" Boolean functions of n Boolean arguments are computed only by noiseless networks with $\Omega\left(2^n/n\right)$ gates
- "Almost all" Boolean functions have bounded redundancy.

Set of Boolean linear functions

- A set of m Boolean functions $f_1(x_1, \cdots, x_n)$, ..., $f_m(x_1, \cdots, x_n)$ is linear if each of the functions is the sum (modulo 2) of some subset of the n Boolean arguments $x_1, \cdots, x_n$.
- "Almost all" sets of n linear functions of n Boolean arguments have bounded redundancy.
  - Similar approach
  - Theorem 4.4

Further readings. . .

- N. Pippenger, “Reliable computation by formulas in the presence of noise”, 1988
- T. Feder, “Reliable computation by networks in the presence of noise”, 1989