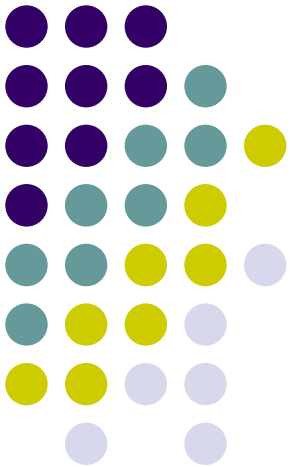
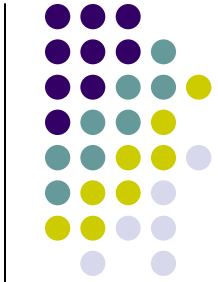
CS256
Applied Theory of Computation

Circuit Complexity II

John E Savage



Overview



- Most Boolean functions are complex
- Simple circuit size lower bound
- The gate elimination method
- Application of the gate elimination method

Most Boolean Functions are Complex

Theorem Let $0 < \varepsilon < 1$. Over the standard basis the fraction of Boolean functions $f : B^n \rightarrow B$ with size complexity $C(f)$ satisfying

$$C(f) \geq (2^n/n)(1-\varepsilon) - 2n^2$$

is at least $\alpha = 1-2^{-\beta}$ where $\beta = (\varepsilon/2)2^n$ when $n \geq N_0$ (see p. 77).

(I.E., when n is large, most Boolean functions on n variables have very large size!)

Most Boolean Functions are Complex

- **Proof** Consider circuits with g gates and no constant inputs. $\leq (g\text{-}1+n)^2$ ways to connect inputs and outputs of other $g\text{-}1$ gates to two inputs of g. 3^g ways to label gates and $g!$ different orderings of g gates ($g! \geq g^g e^{-g}$) Since all orderings give same circuit, there are at most N(g) circuits with g gates where

  $N(g) \leq (3e)^g [(g^2+2gn+n^2)/g]^g \leq (3e)^g(g+2n^2)^g$

  Here $2gn+n^2 \leq 2gn^2$ for $n \geq 2$.

Most Boolean Functions are Complex

Proof (cont.) Summing over $0 \le g \le G$ using $G+1 \le (3e)^G$, we have

$$M(G) \le (G+1)N(G) \le [(3e)^2(G+2n^2)]^G \le (x^x)^{(1/a)}$$

for $x=a(G+2n^2)$, $a=(3e)^2$. It is easily shown that $x^x \le 2^y$ if $x \le y \log_2 y$ and $y \ge 2$.

Let $\beta=(1-\delta)2^n$ and $\gamma=-\delta 2^n$. If $M(G) \le 2^\beta$, at most a fraction $\le 2^{-\gamma}$ of circuits have size $\le G$.

If $G < (2^n/n)(1-\varepsilon) - 2n^2$, $x \le a2^n(1-\varepsilon)/n \le x_0 / \log_2 x_0$ for $x_0 = a2^n(1-\varepsilon/2)$ when $n \ge 2[(1-\varepsilon)/\varepsilon] \log_2[(3e)^2(1-\varepsilon/2)]$. It follows that $M(G) \le (x^x)^{1/a} \le 2^{x0} \le 2^\beta$. ♥

Simple Circuit Size Lower Bounds

The function $f : B^n \rightarrow B$ **depends on variable** x_i if there are values for the remaining variables so that changing x_i changes the value of f.

Theorem Let $f : B^n \rightarrow B$ depend on each of its variables. Over a basis of fan-in r,

$$C(f) \geq (n-1)/(r-1) \text{ and } D(f) \geq \log_r n.$$

Simple Circuit Size Lower Bounds

Theorem Let $f : B^n \rightarrow B$ depend on each of its variables. Over a basis of fan-in r,

$$C(f) \geq (n-1)/(r-1) \text{ and } D(f) \geq \log_r n.$$

Proof Number of edges between gates is at most $rC(f)$ and at least $\geq C(f) + n - 1$ edges because $\geq C(f)$-1 gates have at least one edge directed away from them as does each of n inputs.

A circuit of fan-in r with depth d has at most r^d leaves. Since f depends on n variables, $r^d \geq n$ from which the result follows. ♥

The Gate Elimination Method

Uses induction to provide one of the strongest known lower bounds for complete bases.

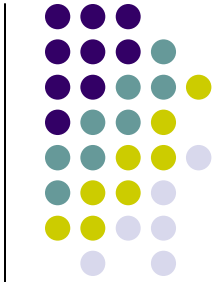
Method:

a) Show that assigning values to a few variables results in function of the same type.

b) Count the number of eliminated gates. After all variables eliminated, function is constant.

c) The circuit size is at least number of gates eliminated, giving lower bound.

The Gate Elimination Method



Definition $Q_{2,3}^{(n)}$ is the set of $f : B^n \rightarrow B$ such that for every pair of variables x_j and x_k f has at least three distinct subfunctions as the two variables range over all four values. Also, there is an x_i such that for some value c_i the subfunction obtained is in $Q_{2,3}^{(n-1)}$.

The Gate Elimination Method

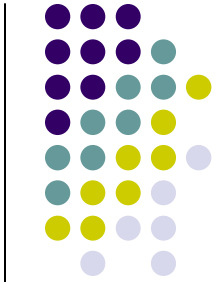
Lemma For n ≥ 3, $Q_{2,3}^{(n)}$ contains $f_{\text{mod } 3,c}^{(n)}$ where $f_{\text{mod } 3,c}^{(n)}(x_1, x_2, \ldots, x_n) = ((y+c) \bmod 3) \bmod 2)$ for c in {0,1,2} where $y = x_1 + x_2 + \ldots + x_n$.

Proof The functions $f_{\text{mod } 3,c}^{(1)}$ for c in {0,1,2} are x_1, $\bar{x}_1$, and 0, respectively. For $n = 2$, $f_{\text{mod } 3,c}^{(2)}$ has value 1 exactly when y =1,0,2 for c in {0,1,2}, respectively. We show that property holds for $n \geq 3$.

The Gate Elimination Method

Proof (cont.) In $f_{\text{mod } 3,c}^{(n)}(x_1, x_2, \ldots, x_n)$ fix any two variables and let y* be the sum of the remaining *n*-2 variables and c* be the sum of c and the values of the two fixed variables. Then, ((y+c)mod 3)mod 2)=(((y* mod 3 + c* mod 3) mod 3)mod 2). Since the value of y* mod 3 is in {0,1,2} and c* mod 3 has values in {0,1,2}, from above (((y* mod 3 + c* mod 3) mod 3) mod 2) has one of 3 different functions. With c_i =0, the resulting subfunction is in $Q_{2,3}^{(n)}$.♥

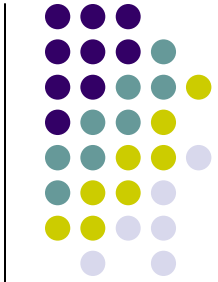
The Gate Elimination Method



Theorem Over the basis of all Boolean functions on 2 inputs, f in $Q_{2,3}^{(n)}$ for $n \geq 3$ has $C(f) \geq 2n-3$.

Proof f depends on each of its variables because if not, there is an x_i such that f doesn't depend on it. If so, then picking any second variable, f has at most two subfunctions, a contradiction.

The Gate Elimination Method



Proof (cont.) We show that fan-out from some input is at least two. Consider gate *g* such that the path to the output from it is longest. Both of its inputs are from variables. If each input to that gate has fan-out one, f has at most two subfunctions on these input variables, a contradiction. Thus, for n=3, there are at least 4 edges from inputs from which the lower bound follows from the simple linear lower bound.

Assume that $C(f) \geq 2n-3$ for $n \leq k$. For $n = k+1$, fix an input with fan-out 2, thereby deleting two gates.♥

The Gate Elimination Method

- We derive a lower bound using this method for the storage access function $f_{SA}(a_{k-1}, \ldots, a_0, x_{n-1}, \ldots, x_0) = x_{|a|}$ where $|a|$ is the integer specified by $(a_{k-1}, \ldots, a_0)$.
- This function can be implemented with inputs having fan-out one. Thus, we have to look closely at the functions used in the basis.

The Gate Elimination Method

Definition $f : B^{n+k} \rightarrow B$ belongs to $F_s^{n,k}$, $2^k \geq n$ if for some set S subset of {0,1,...,n-1}, |S| = s if the following holds for |a| in S.

$$f(a_{k-1}, \ldots, a_0, x_{n-1}, \ldots, x_0) = x_{|a|}$$

Clearly f_{SA} is a member of $F_n^{n,k}$. We show that every function in $F_s^{n,k}$ has circuit size at least $2s-2$.

Let B_2 be the basis containing all Boolean functions on two inputs.

The Gate Elimination Method

Theorem Let $f : B^{n+k} \rightarrow B$ be in $F_s^{n,k}$, $2^k \geq n$. Over B_2, $C(f) \geq 2s\text{-}2$.

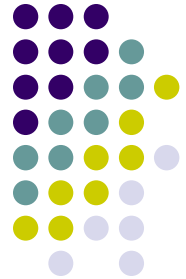
Proof By induction on s. Base case s=1 is obvious. Suppose hypothesis holds for $s = s'\text{-}1$. We show that it holds for $s = s'$. Cases:

a) For i in S, x_i has fan-out 2. Replacing x_i by right constant gives f^* in the class with $s = s'\text{-}1$ and eliminates two gates.

$$C(f) \geq 2 + C(f^*) \geq 2 + 2s'\text{-}2 = 2s\text{ -}2$$

Note: all gates in B_2 are either AND-type (computing AND or OR) or XOR-type (computing x_i XOR g XOR c)

The Gate Elimination Method



b) For *i* in S, x_i has fan-out 1 to AND-type gate computing function $(x_i^a$ AND $g^b)^c$. Setting x_i^a to 0 makes gate constant, eliminating it & ≥ *1* successor.

c) For *i* in S, x_i has fan-out 1 to XOR-type gate computing G (x_i XOR g XOR c) for some function g of the inputs. This XOR-type gate is not the output. By fixing x_i eliminate *i* from S. Since |*a*| is not in S - {*i* }, we can set x_i to any value or function without affecting membership of *f* with x_i fixed in the class. Thus, set x_i to g causing the output of G to be a constant and eliminating at least two gates. ♥