Small-depth circuits, a revisit

Johan Håstad

KTH Numerical Analysis
and Computer Science

St Petersburg, May 23, 2016

Ask questions

Do ask questions during the talk.

I am not fond of speaking for too long on my own.

The plan

A long time ago, I wrote a thesis on circuit complexity.

Some open problems from the thesis were recently solved by very similar methods.

What happened?

My opinion

Memories are “adjustable”.

If you think about something again and again the memory changes.

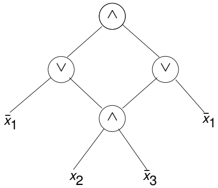
Nicer and more general ways of saying something are adopted.

“I was thinking about it this way all along but just did not write it this way”.

On a few accounts I know I am guilty of this.

Basic definition

A circuit is a directed acyclic graph from inputs to one output with n inputs.

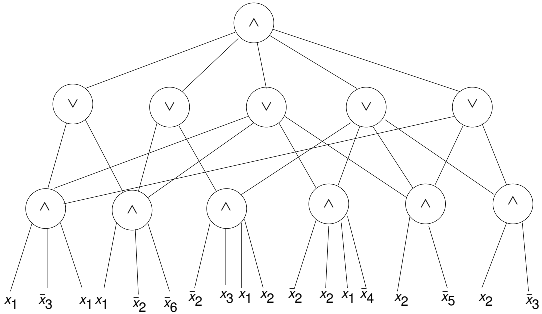


Size: Number of gates, $S = 4$

Depth: Longest path from input to output, $d = 3$

Small-depth circuits

Unbounded fanin circuits with $\wedge$ and $\vee$-gates in alternating layers. Neighboring gates of same type can be collapsed.



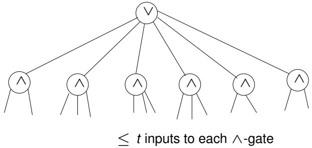
Classical questions

What size is needed to computer parity (exact or approximate)?

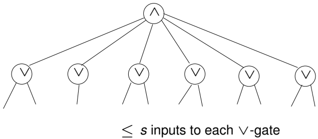
Is depth k more powerful than depth $k-1$ (and say polynomial size)?

Something easy? to understand

Depth 2. A t-DNF



and an s-CNF



If f computed by one then $\neg f$ computed by the other and thus these are equally hard to study.

Computing parity in small depth

For depth 2 parity requires size $1 + 2^{n-1}$ and bottom fanin n both as a CNF and DNF.

Not difficult to establish exact bounds.

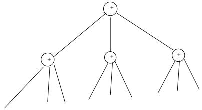
Parity in depth 4

A parity tree of depth 2 of fan-out $\sqrt{n}$.

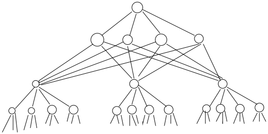
Replace each gate by a depth two circuit of size $2^{\sqrt{n}}$.

Circuit of depth 4 and size $n2^{\sqrt{n}}$.

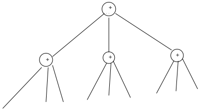
In pictures



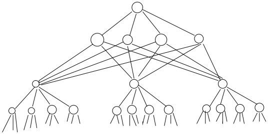
turns into



In pictures

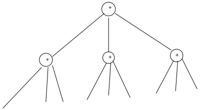


turns into

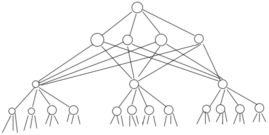


Ignoring negations, which causes a factor 2 blowup.

In pictures



turns into



Ignoring negations, which causes a factor 2 blowup.

Take CNF for top gate and DNF for second level. Adjacent levels of or-gates and we can decrease depth to 3.

Higher depth

Parity tree of depth k and fan-out $n^{1/k}$.

CNFs on odd levels, DNFs on even levels.

Depth $k+1$ and size $n2^{n^{1/k}}$.

Parity result of 1986

The final theorem after work by Furst, Saxe, and Sipser, Ajtai and Yao.

Theorem [H86] To compute parity of n variables in depth d you need size

$$2^{\Omega(n^{1/(d-1)})}.$$

Parity result of 1986

The final theorem after work by Furst, Saxe, and Sipser, Ajtai and Yao.

Theorem [H86] To compute parity of n variables in depth d you need size

$$2^{\Omega(n^{1/(d-1)})}.$$

In fact (joint with Ravi Boppana) with size smaller than this you can only agree with parity for a fraction

$$\frac{1}{2} + 2^{-\Omega(n^{1/(d-1)})}$$

of the inputs.

Best possible?

For exact computing best possible up to the implied constant.

For correlation, strangely not.

Best possible?

For exact computing best possible up to the implied constant.

For correlation, strangely not.

For polynomial size, Ajtai's result gave better bounds for correlation and this was something strange already then.

More recent results, 2014

We will outline an argument that a circuit of size S and depth d can only agree with parity on a fraction

$$\frac{1}{2} + 2^{-\Omega(n/(\log S)^{d-1})}$$

of the inputs.

Proved independently by Impagliazzo, Matthews and Paturi.

The hierarchy theorem

Strengthening work of Sipser and Yao we proved

Theorem [H86] There is a function, f_d computable by a read-once formula of depth d that requires size

$$2^{\Omega(n^{\Omega(1/d)})}$$

to be computed by depth $d-1$.

The hierarchy theorem

Strengthening work of Sipser and Yao we proved

Theorem [H86] There is a function, f_d computable by a read-once formula of depth d that requires size

$$2^{\Omega(n^{\Omega(1/d)})}$$

to be computed by depth $d-1$.

Open question: How about non-trivial agreement?

The hierarchy theorem

Strengthening work of Sipser and Yao we proved

Theorem [H86] There is a function, f_d computable by a read-once formula of depth d that requires size

$$2^{\Omega(n^{\Omega(1/d)})}$$

to be computed by depth $d-1$.

Open question: How about non-trivial agreement?

I do not even think I was a 100% convinced that the strengthening was true, at least not for read-once formulas.

More recent results, 2015

Rossman, Servedio and Tan prove that (for a different function f_d) the agreement can be at most

$$\frac{1}{2} + n^{-\Omega(1/d)}.$$

I extend this from $d = \sqrt{\log n} / \log \log n$ to $\log n / \log \log n$ and make the proof more succinct.

Rest of talk

Discussing original proofs and what adjustments were needed for the more modern results.

Restrictions

Sipser [S83]: Randomly give values to most of the variables.

Restrictions

Sipser [S83]: Randomly give values to most of the variables.

Formally: $\rho \in R_p$ for each variable x_i independently:

Keep it is a variable with probability p, otherwise fix it to 0 and 1 with equal probability, $(1-p)/2$.

Notation $\rho(x_i) = 0, 1, *$.

What restrictions do

Restrictions simplify small-depth circuit.

Functions that survive restrictions are hard to compute by small-depth circuits.

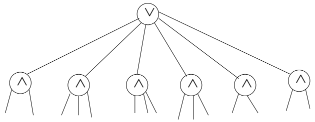
Parity survives any restriction but for other functions we need more sensitive spaces of restrictions.

The switching lemma

After preliminary work by Yao with more complicated notions.

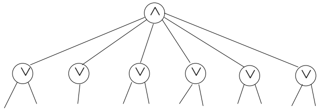
Lemma [H86] Any depth two circuit which is a $\vee$ of $\wedge$'s each of which is size $\leq t$ can, when hit with a random $\rho \in R_p$, with probability at least $1 - (5pt)^s$, be converted to a depth two circuit which is a $\wedge$ of $\vee$'s each of which is of size $\leq s$.

A picture



$\leq$ *t* inputs to each $\wedge$-gate

turns into



$\leq$ *s* inputs to each $\vee$-gate

with probability $1 - (5pt)^s$ by $\rho \in R_p$.

And the other way around.

Proofs of the switching lemma

Original proof by me through a labeling argument working with conditioning of clauses of the formula.

Ravi Boppana suggested to write it with arbitrary conditioning and induction. Which I adopted and forgot that this was Ravi's suggestion.

Sasha Razborov later showed how to write it as a labeling argument.

Proofs of the switching lemma

Original proof by me through a labeling argument working with conditioning of clauses of the formula.

Ravi Boppana suggested to write it with arbitrary conditioning and induction. Which I adopted and forgot that this was Ravi's suggestion.

Sasha Razborov later showed how to write it as a labeling argument.

I think the arguments are (essentially) the same but the induction formalism gives shorter proofs with less notation.

Variants in statement

Originally I proved proved that each minterm is of size at most s.

Cai suggested that it is better to say that the depth of a decision tree is at most t.

Originally proved also under conditions of the form $F\lceil_\rho \equiv 1$, (including Boppana).

I now prefer conditioning $\rho \in \Delta$ for a downward closed set Δ.

Downward closed sets

If $\rho \in \Delta$ and $\rho(x_i) = *$, then changing this value to 0 or 1 does not make ρ leave Δ. Examples

- The set of restrictions forcing F to the constant 1.
- The set of restrictions that give the value $*$ to at most pn variables.
- The set of restrictions that make C possible to compute by a decision tree of depth at most 7.

Proof summary

Let $C = \wedge_{i=1}^{m} C_i$ where each $|C_i| \le t$. Want to prove.

$$Pr[depth(C\lceil_\rho) \ge s | \rho \in \Delta] \le (5pt)^s,$$

by induction over m.

If $C_1\lceil_\rho \equiv 1$ we stick it into the conditioning and use induction.

If $C_1\lceil_\rho \not\equiv 1$ we put the variables in C_1 which are given the value $*$ by ρ into the decision tree and apply induction.

The key sub-lemma

For any set of variables Y appearing in C_1.

$$Pr[\rho(Y) = * \mid C_1\lceil_\rho \not\equiv 1 \wedge \rho \in \Delta] \leq (\frac{2p}{1+p})^{|Y|}.$$

Belonging to Δ does not bias coordinates towards being $*$.

The key sub-lemma

For any set of variables Y appearing in C_1.

$$Pr[\rho(Y) = * \mid C_1\lceil_\rho \not\equiv 1 \wedge \rho \in \Delta] \leq (\frac{2p}{1+p})^{|Y|}.$$

Belonging to Δ does not bias coordinates towards being $*$.

Proof of sub-lemma: Take any ρ contributing to the event and change its value on Y to other values consistent with $C_1\lceil_\rho \not\equiv 1$. This gives restrictions satisfying the conditioning.

The key fact

We need

$$\frac{Pr[\rho(x_i) = *]}{Pr[\rho(x_i) = 0 \vee \rho(x_i) = *]}$$

and

$$\frac{Pr[\rho(x_i) = *]}{Pr[\rho(x_i) = 1 \vee \rho(x_i) = *]}$$

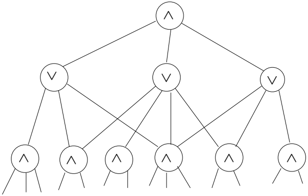
to be small. For $\rho \in R_p$ these are $\frac{2p}{1+p}$, the number that shows up in key sub-lemma.

Switching gives parity lower bound

Induction with $p = n^{-1/(d-1)}$ and $s = t = \frac{1}{10} n^{1/(d-1)}$.

Each restriction wipes out one level.

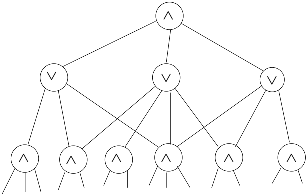
In pictures, I



bottom fanin $\leq t$

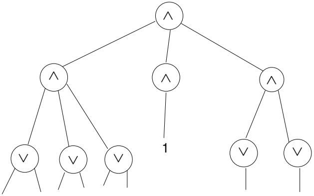
Apply $\rho \in R_p$ and use lemma on each depth 2 sub-circuit.

In pictures, I



bottom fanin $\leq t$

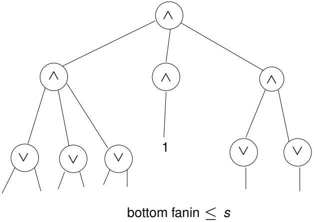
Apply $\rho \in R_p$ and use lemma on each depth 2 sub-circuit.



bottom fanin $\leq s$

In pictures, II

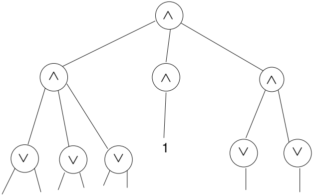
After switching we have



and we make shortcuts

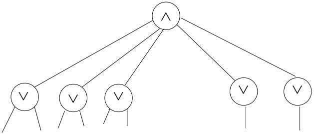
In pictures, II

After switching we have



bottom fanin $\leq s$

and we make shortcuts



bottom fanin $\leq s$

Punch line, switching

Repeat until circuit has depth 2.

When reduced to a decision tree of not full depth there is no correlation with parity.

Parameters

If size of the circuit is S, the probability to fail at least some switching is $S2^{-t}$.

Make sure that the expected number of variables remaining in the end is $2t$.

The probability of agreeing with parity is at most $\frac{1}{2} + S2^{-t} + 2^{-ct}$.

Clearly best possible!?

The estimate of the switching lemma is essentially sharp.

Need $p \leq \frac{1}{t}$ and cannot get better than exponential decay in s.

Clearly best possible!?

The estimate of the switching lemma is essentially sharp.

Need $p \leq \frac{1}{t}$ and cannot get better than exponential decay in s.

Correlation $\frac{1}{2} + 2^{-r}$ requires failure probabilities 2^{-r} and thus $s \approx r$ and thus we need to have $t \approx r$ and $p \approx \frac{1}{r}$.

Only at most nr^{2-d} variables remain.

Clearly best possible!?

The estimate of the switching lemma is essentially sharp.

Need $p \leq \frac{1}{t}$ and cannot get better than exponential decay in s.

Correlation $\frac{1}{2} + 2^{-r}$ requires failure probabilities 2^{-r} and thus $s \approx r$ and thus we need to have $t \approx r$ and $p \approx \frac{1}{r}$.

Only at most nr^{2-d} variables remain.

I was stuck. I could not see a way around this in 1986 and gave up.

Looking more closely

With probability $S2^{-s}$ we need to handle the event that an individual depth-two circuit has some path in its decision tree of length s.

Looking more closely

With probability $S2^{-s}$ we need to handle the event that an individual depth-two circuit has some path in its decision tree of length s.

Usually a single path of length s appearing in a single decision tree being constructed.

The failure is extremely local.

Getting stronger correlation

Apply restriction ρ.

- Go over depth two circuits, D_i, one by one.
- If depth of decision tree of $D_i\lceil_\rho$ is at most $10 \log S$, switch it.
- If some path of the decision tree of $D_i\lceil_\rho$ is at least $10 \log S$ fix the variables along *this* path.

Apply induction on the number of depth-2 sub-circuits.

Getting stronger correlation

Apply restriction ρ.

- Go over depth two circuits, D_i, one by one.
- If depth of decision tree of $D_i\lceil_\rho$ is at most $10 \log S$, switch it.
- If some path of the decision tree of $D_i\lceil_\rho$ is at least $10 \log S$ fix the variables along *this* path.

Apply induction on the number of depth-2 sub-circuits.

As the proof of the Switching lemma but on one higher level.

Getting stronger correlation

Apply restriction ρ.

- Go over depth two circuits, D_i, one by one.
- If depth of decision tree of $D_i\lceil_\rho$ is at most $10 \log S$, switch it.
- If some path of the decision tree of $D_i\lceil_\rho$ is at least $10 \log S$ fix the variables along *this* path.

Apply induction on the number of depth-2 sub-circuits.

As the proof of the Switching lemma but on one higher level.

The Switching lemma does for this argument what the “key sub-lemma” did for the Switching lemma.

The structure of overall proof

1. At stage i we have a circuit of depth $d - i$ with bottom fanin $10 \log S$ and size S.
2. We apply a restriction with $p = (c \log S)^{-1}$.
3. We switch each sub-circuit of depth 2 maintaining fanin $10 \log S$. If needed we fix some extra variables.

The probability of being forced to fix the value of k extra variables is 2^{-ck}.

To apply the induction

Need to make sure that the conditioning is of the proper form, i.e. downward closed.

Intuitive reasons

- A successful switch is a downward closed condition.
- If we get a long path in a decision tree, we fix the values of all “touched” variables.

A subtle point

The property "the decision tree of $C\lceil_\rho$ created by the proof of the switching lemma is of depth at most s" is is not a downward closed property.

However, "$C\lceil_\rho$ has a decision tree of depth at most s" is a downward closed property, and this is enough.

The correlation of parity theorem

In the end we get

Theorem Let f be computed by a depth d circuit of size S. Then

$$Pr[f(x) = parity(x)] \leq \frac{1}{2} + 2^{-\Omega(n/(\log S)^{d-1})}.$$

The mental lesson

Maybe something is likely to go wrong, but maybe the price to pay to fix it is much smaller than you think at first.

Do not panic!

Try again!

Proving hierarchy theorem

We have two circuits.

The defining circuit Depth d and small (probably size n). Computes f_d and has known structure.

The competing circuit Depth $d-1$ and large. Unknown structure, except possibly for small bottom fanin. Should not compute f_d.

If the size of the competing circuit is S and we are doing switching we probably have bottom fanin $T \approx \log S$.

An obstacle

We need to have large bottom fanin of the defining circuit. At least as large as the competing circuit. We need to maintain this property in each step of the induction.

An obstacle

We need to have large bottom fanin of the defining circuit. At least as large as the competing circuit. We need to maintain this property in each step of the induction.

Any gate with fanin T takes its “favorite” value (true for or-gates, false for and-gates) with probability $1 - 2^{-T}$.

An obstacle

We need to have large bottom fanin of the defining circuit. At least as large as the competing circuit. We need to maintain this property in each step of the induction.

Any gate with fanin T takes its "favorite" value (true for or-gates, false for and-gates) with probability $1 - 2^{-T}$.

Replacing gates next to the input by their favorite values we get a constant that equals the value of the defining circuit with probability at least $1 - n2^{-T} = 1 - n/S^c$.

An obstacle

We need to have large bottom fanin of the defining circuit. At least as large as the competing circuit. We need to maintain this property in each step of the induction.

Any gate with fanin T takes its “favorite” value (true for or-gates, false for and-gates) with probability $1 - 2^{-T}$.

Replacing gates next to the input by their favorite values we get a constant that equals the value of the defining circuit with probability at least $1 - n2^{-T} = 1 - n/S^c$.

Superpolynomial lower bounds seems to require a defining circuit that is well approximated by a constant.

An obstacle

We need to have large bottom fanin of the defining circuit. At least as large as the competing circuit. We need to maintain this property in each step of the induction.

Any gate with fanin T takes its "favorite" value (true for or-gates, false for and-gates) with probability $1 - 2^{-T}$.

Replacing gates next to the input by their favorite values we get a constant that equals the value of the defining circuit with probability at least $1 - n2^{-T} = 1 - n/S^c$.

Superpolynomial lower bounds seems to require a defining circuit that is well approximated by a constant.

It seems completely impossible to prove average-case lower bounds in the hierarchy setting by a switching lemma?!

As far as I remember

I think this was the end of my thinking on the subject in the 1980'ies.

Checking again

The two statements.

1. The defining circuit needs bottom fanin at least $T = \Omega(\log S)$.
2. Any circuit of size n and bottom fanin T takes it favorite value with probability $1 - n2^{-T}$.

Checking again

The two statements.

1. The defining circuit needs bottom fanin at least $T = \Omega(\log S)$.
2. Any circuit of size n and bottom fanin T takes it favorite value with probability $1 - n2^{-T}$.

Do we need to have unbiased inputs?

Checking again

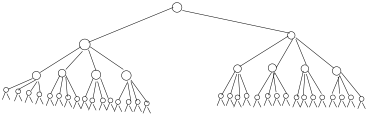
The two statements.

1. The defining circuit needs bottom fanin at least $T = \Omega(\log S)$.
2. Any circuit of size n and bottom fanin T takes it favorite value with probability $1 - n2^{-T}$.

Do we need to have unbiased inputs?

In fact not. Original inputs need to be unbiased to get average case, but we can introduce intermediate variables to denote more complicated objects.

The function F_d



Computed by a read-once formula.

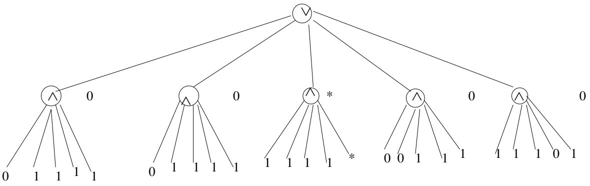
Top fanin 2^{2m}.

Middle level fan-ins $\Theta(m2^{2m})$.

Bottom fanin $\Theta(m2^m)$.

Inputs are $1 - 2^{-m}$ biased and given by the conjunction of m unbiased variables.

Independently for each depth 2 circuit of defining circuit.



Set value of gate v to a biased variable b_v which is 0 with probability $1-q$ and otherwise *.
Make each variable feeding into v equal to 1 with probability $1-q$ and otherwise b_v.

Sanity check

Both

$$\frac{Pr[\rho(x_i) = *]}{Pr[\rho(x_i) = 0 \vee \rho(x_i) = *]}$$

and

$$\frac{Pr[\rho(x_i) = *]}{Pr[\rho(x_i) = 1 \vee \rho(x_i) = *]}$$

are about q even conditioning on a downward closed Δ.

Sanity check

Both

$$\frac{Pr[\rho(x_i) = *]}{Pr[\rho(x_i) = 0 \vee \rho(x_i) = *]}$$

and

$$\frac{Pr[\rho(x_i) = *]}{Pr[\rho(x_i) = 1 \vee \rho(x_i) = *]}$$

are about q even conditioning on a downward closed Δ.

For the first change b_v from $*$ to 0, for the second only x_i.

Items to worry about

- Will gate v really take value b_v?
- If we find one $*$ this biases other variables to $*$.
- Handing out values with too much dependence is dangerous for the proof of the Switching lemma.

The second item is true of variables in the same gate.

Dealing with problems in 1986

If the fanin of the gate is T.

If we set qT large enough, v is very likely to take the value b_v.

Dealing with problems in 1986

If the fanin of the gate is T.

If we set qT large enough, v is very likely to take the value b_v.

Forces a non-uniformly picked input.

Once we have applied ρ we had an additional step of fixing all but one variable in each block.

Dealing with problems in 1986

If the fanin of the gate is T.

If we set qT large enough, v is very likely to take the value b_v.

Forces a non-uniformly picked input.

Once we have applied ρ we had an additional step of fixing all but one variable in each block.

Probably creates a non-uniformly picked input.

Solution by [RST]

Allow the gate not to take the value b_v. Make sure that this does not destroy the defining circuit too much.

Identify all variables given the value b_v in the same block with a the same new variable. Need to be careful to get the correct distribution. “Projections”

A delicate game to make a biased selection of b_v give the independent distribution overall.

Properties to balance

1. Not destroying the defining circuit.
2. Making the input uniformly random.
3. Making it possible to prove the switching lemma.

The more independent we pick various parts of the restriction, the easier is 3 and the harder is 1.

The condition 2 needs to be true by definition and leaves little choice.

New proof vs [RST]

Rather technical.

Focus more directly on not destroying the defining circuit.

Proof of Switching lemma with induction rather than labeling.

The key sub-lemma of the Switching lemma does not require much.

For any $d \leq c \log n / \log\log n$ we have.

Theorem There is a function F_d computed by a read-once depth d formula such that for any circuit, C, of size $2^{O(n^{1/5d})}$ and depth at most $d-1$ we have

$$Pr[C(x) = F_d(x)] \leq \frac{1}{2} + n^{-1/8d}.$$

Rossman, Servedio and Tan had this for $d \leq \sqrt{\log n} / \log\log n$.

Exponentially small?

Can we make the correlation exponentially small?

Exponentially small?

Can we make the correlation exponentially small?

Not possible for a read-once formula as one input of the top gate is at least $\frac{1}{2} + \frac{1}{n}$ correlated with the output.

Exponentially small?

Can we make the correlation exponentially small?

Not possible for a read-once formula as one input of the top gate is at least $\frac{1}{2} + \frac{1}{n}$ correlated with the output.

At first I thought it would be extremely difficult to define a suitable function but making the talk I became more optimistic.

A final mental lesson

If you each day firmly believe that what want to prove is not only true but provable by the ideas you have at hand, then you do well.

When you are correct you are much more likely to find the proof.

When you are wrong you would not have found the proof anyhow.

A final mental lesson

If you each day firmly believe that what want to prove is not only true but provable by the ideas you have at hand, then you do well.

When you are correct you are much more likely to find the proof.

When you are wrong you would not have found the proof anyhow.

Can you each day really convince yourself and still remain sane?

Induction vs labeling

For the switching lemma I think induction is better than labeling.

Does require some self-confidence when thinking about conditioning.

I had some incorrect proofs for both theorems mentioned in this talk at first.

The End