Connections between QBF proof complexity and circuit complexity

Olaf Beyersdorff

Friedrich Schiller University Jena, Germany

Hard problems everywhere?

Hard Problem 1: Circuit complexity

- show explicit circuit size lower bounds for (strong) circuit classes

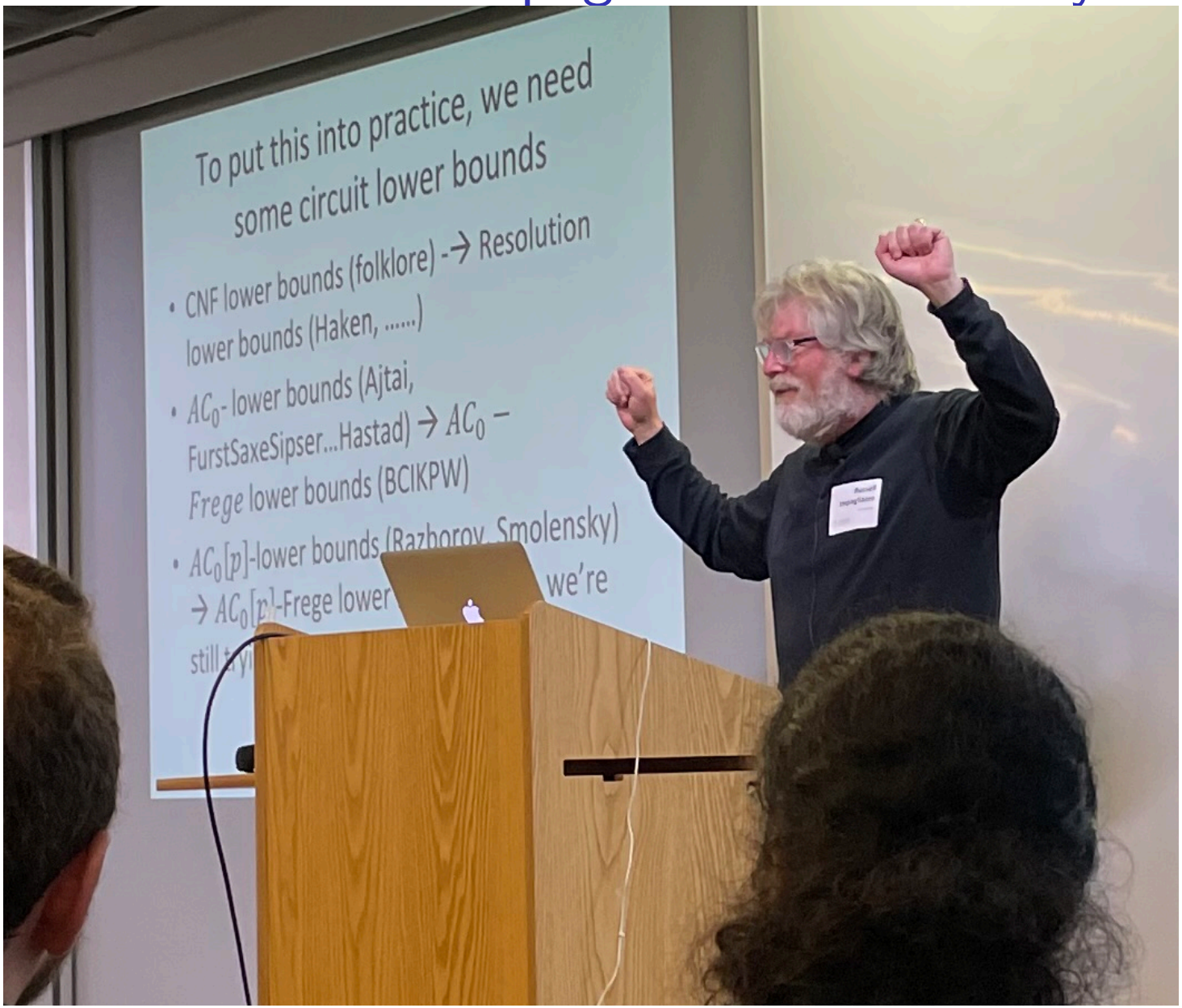
Hard Problem 2: Proof complexity

- show proof size lower bounds for (strong) proof systems

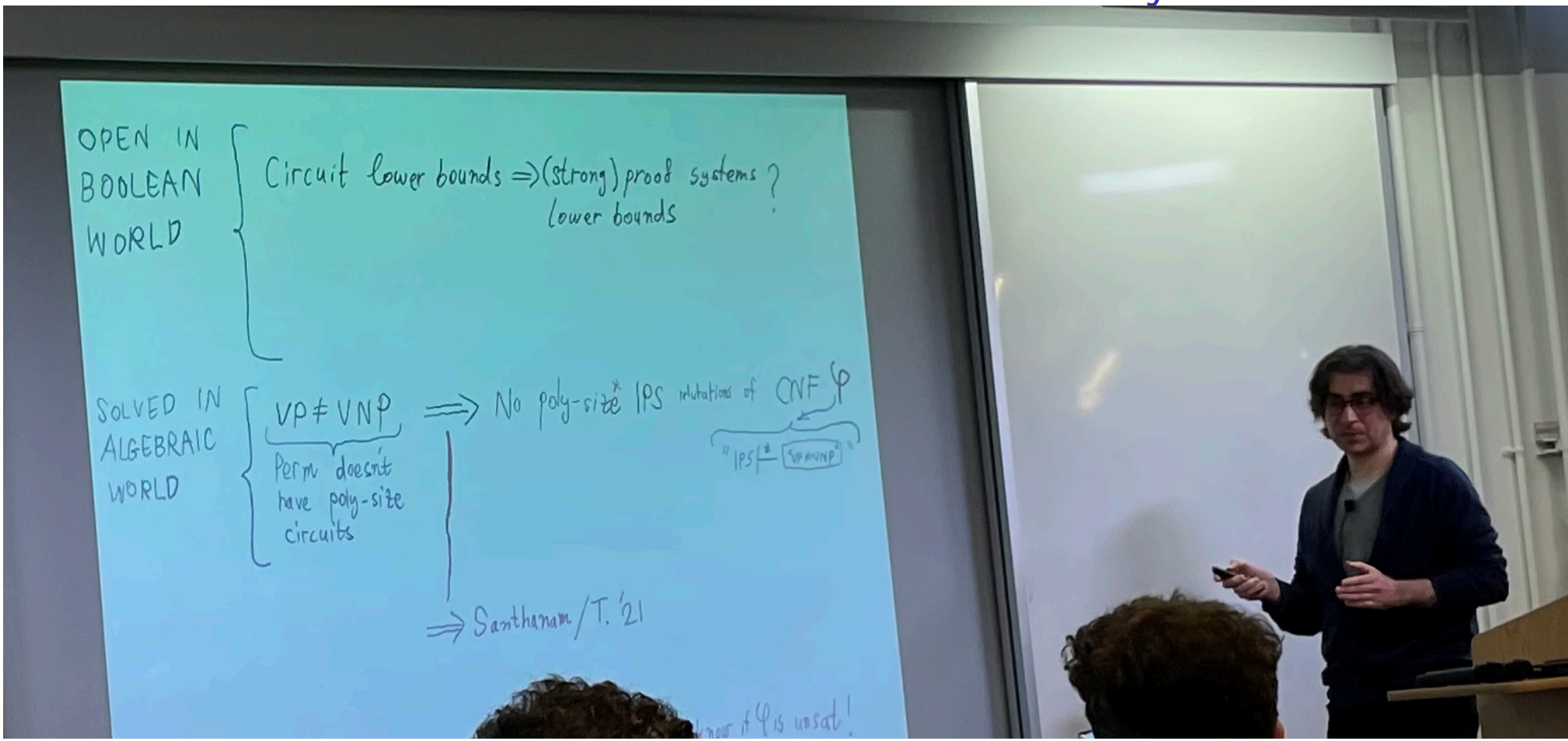
(Hard) Question

- Is there any connection between Problems 1 and 2?

Russel Impagliazzo this Monday



Iddo Tzameret this Wednesday



Such a connection has often been postulated

The correspondence between circuit classes and proof systems has not only been fruitful in developing ideas for new proof systems. It has also been the avenue for applying circuit lower bound techniques to propositional proofs. Some of the major progress of the last decade building on the original insight due to Ajtai, has been in achieving lower bounds for Frege proof systems and their extensions.
In general, the intuition for this approach is that any tautology that needs to use in its proof some concept that is not representable in complexity class $\mathcal{C}$ will not be efficiently provable in $\mathcal{C}$-Frege.

Paul Beame & Toni Pitassi 2001

The proof complexity theme song

You say you work on resolution
Well, you know, we all want a lower bound
You tell me you'd add substitution
Well, you know, first you gotta prove it sound

...

You say you can prove Pigeonhole
Well, you know, hard examples are hard to find
Though bounds for circuits play a role
Well, you know, this connection isn't well-defined

...

Jan Johannsen & Antonina Kolokolova

Proofs vs circuits: what's the question exactly?

A formal connection?

- general belief: there is a connection between lower bounds for proof systems working on $\mathcal{C}$ circuits and lower bounds for $\mathcal{C}$
- has not been made formal yet

Examples

- Are lower bounds for P/poly and lower bounds for EF related?
- same for $\mathrm{AC}^0[p]$ vs $\mathrm{AC}^0[p]$-Frege ...

Resolution and feasible interpolation

- imports lower bounds for monotone circuits

Algebraic proof systems

- connections between algebraic proof systems and lower bounds for algebraic circuits

This talk: The situation in QBF

Quantified Boolean Formulas (QBF)

- PSPACE-complete problem
- extensive work on QBF solving and proof complexity in the last two decades
- we work with fully quantified prenex formulas , e.g.

$$\underbrace{\exists x \forall u \exists t}_{\text{quantifier prefix}} \quad \underbrace{(x \vee u \vee t) \wedge (\neg x \vee \neg u \vee t) \wedge \neg t}_{\text{CNF matrix}}$$

- Such QBFs are either true or false.
- We consider refutation calculi for false QBFs.

Game semantics of QBFs

2-player game between $\exists$ and $\forall$

- following to the prefix, set variables to 0/1
- $\forall$ wins if a clause gets falsified, otherwise $\exists$ wins.

Game semantics of QBFs

2-player game between $\exists$ and $\forall$

- following to the prefix, set variables to 0/1
- $\forall$ wins if a clause gets falsified, otherwise $\exists$ wins.
- Example

$$\exists x \forall u \exists t \ (x \vee u \vee t) \wedge (\neg x \vee \neg u \vee t) \wedge \neg t$$

Game semantics of QBFs

2-player game between $\exists$ and $\forall$

- following to the prefix, set variables to 0/1
- $\forall$ wins if a clause gets falsified, otherwise $\exists$ wins.
- Example

$$\exists x \forall u \exists t \; (x \vee u \vee t) \wedge (\neg x \vee \neg u \vee t) \wedge \neg t$$

- $\exists$ sets $x = 1$

Game semantics of QBFs

2-player game between $\exists$ and $\forall$

- following to the prefix, set variables to 0/1
- $\forall$ wins if a clause gets falsified, otherwise $\exists$ wins.
- Example

$$\exists x \forall u \exists t \; (x \vee u \vee t) \wedge (\neg x \vee \neg u \vee t) \wedge \neg t$$

- $\exists$ sets $x = 1$
- $\forall$ sets $u = 1$

Game semantics of QBFs

2-player game between $\exists$ and $\forall$

- following to the prefix, set variables to 0/1
- $\forall$ wins if a clause gets falsified, otherwise $\exists$ wins.
- Example

$$\exists x \forall u \exists t \; (x \vee u \vee t) \wedge (\neg x \vee \neg u \vee t) \wedge \neg t$$

- $\exists$ sets $x = 1$
- $\forall$ sets $u = 1$
- $\exists$ sets $t = 1$ and loses

A core QBF system: QU-Resolution

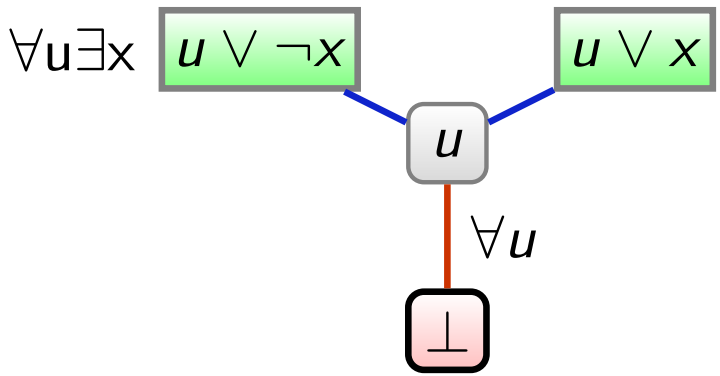
= Resolution + $\forall$-reduction [Kleine Büning et al. 95, V. Gelder 12]

Rules

- Resolution: $\dfrac{x \vee C \qquad \neg x \vee D}{C \vee D}$ ($C \vee D$ is not tautological.)
- $\forall$-Reduction: $\dfrac{C \vee u}{C}$ (u universally quantified)

  C does not contain variables right of u in the quantifier prefix.

Example



From propositional proof systems to QBF

A general $\forall$red rule

- Fix a prenex QBF Φ.
- Let $F(\vec{x}, u)$ be a propositional line in a refutation of Φ, where u is universal with innermost quant. level in F

$$\frac{F(\vec{x}, u)}{F(\vec{x}, 0)} \qquad \frac{F(\vec{x}, u)}{F(\vec{x}, 1)} \qquad (\forall red)$$

QBF proof systems

For any 'natural' line-based propositional proof system P define the QBF proof system $Q\text{-}P$ by adding $\forall$red to the rules of P.

Proposition (B., Bonacina & Chew 16)

$Q\text{-}P$ is sound and complete for QBF.

From propositional proof systems to QBF

A general $\forall$red rule

- Fix a prenex QBF Φ.
- Let $F(\vec{x}, u)$ be a propositional line in a refutation of Φ, where u is universal with innermost quant. level in F

$$\frac{F(\vec{x}, u)}{F(\vec{x}, 0)} \qquad \frac{F(\vec{x}, u)}{F(\vec{x}, 1)} \qquad (\forall red)$$

QBF proof systems

For any 'natural' line-based propositional proof system P define the QBF proof system Q-P by adding $\forall$red to the rules of P.

Remark

For $P =$ Resolution this exactly yields QU-Resolution.

Key example: Frege systems

Frege systems

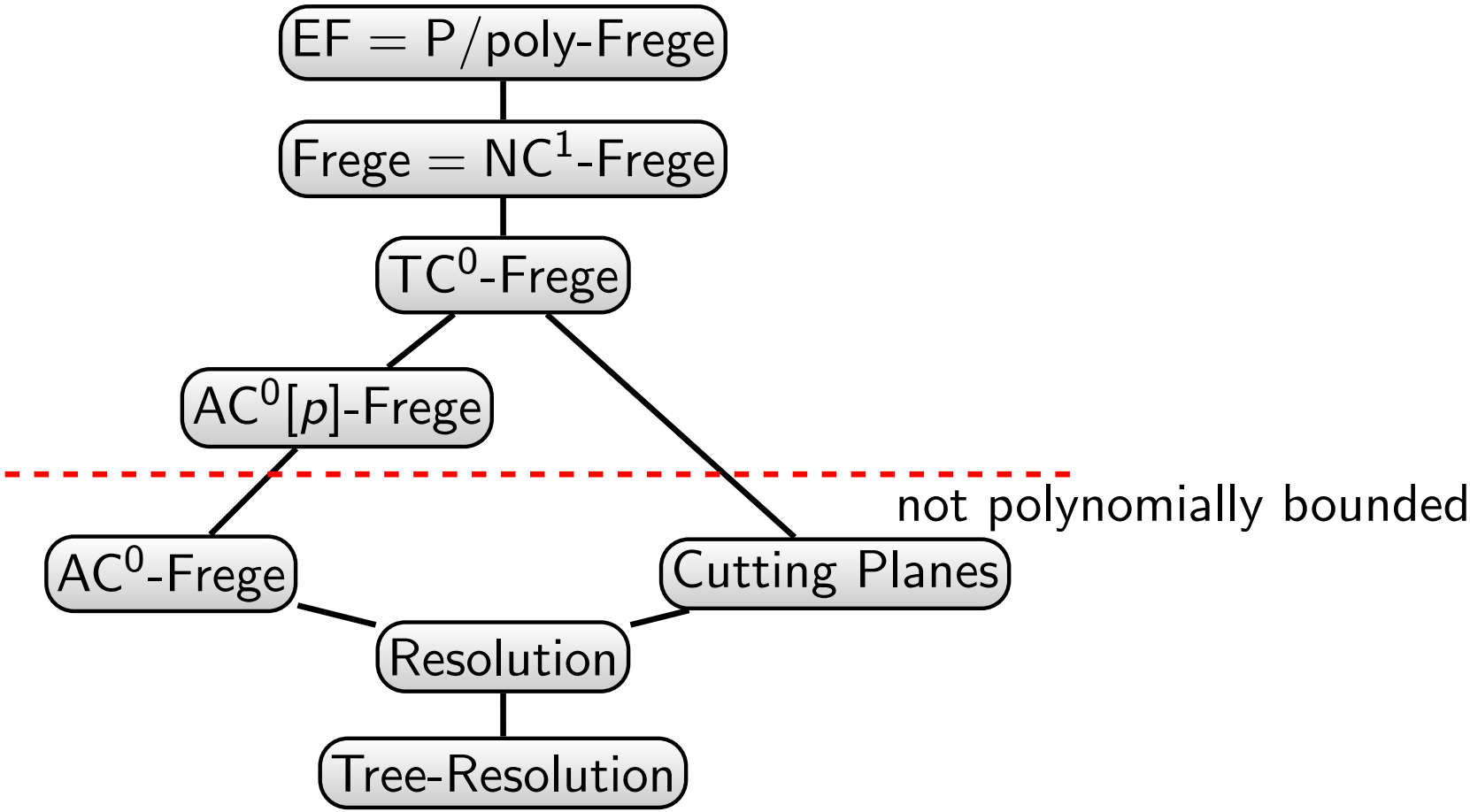
- Hilbert-type systems
- use axiom schemes and rules, e.g. modus ponens $\frac{A \quad A \to B}{B}$

A hierarchy of Frege systems

$\mathcal{C}$-Frege where $\mathcal{C}$ is a circuit class restricting the formulas allowed in the Frege system, e.g.

- AC^0-Frege = bounded-depth Frege
- $\mathrm{AC}^0[p]$-Frege = bounded-depth Frege with mod p gates for a prime p
- TC^0-Frege = bounded-depth Frege with threshold gates

Important propositional proof systems



Genuine QBF lower bounds

Propositional hardness transfers to QBF

- If $\phi_n(\vec{x})$ is hard for P, then $\exists\vec{x}\,\phi_n(\vec{x})$ is hard for Q-P.
- propositional hardness: not the phenomenon we want to study.

Genuine QBF hardness

- in Q-P: just count the number of $\forall$red steps
- can be modelled precisely by allowing NP oracles in QBF proofs [Chen 16; B., Hinde & Pich 17]

QBF proof systems with NP oracles

The QBF system $Q\text{-}P^{NP}$ has the rules:

- of the propositional system P
- $\forall$-reduction
- $\dfrac{C_1 \ \ldots \ C_\ell}{D}$ for any ℓ,
  where $\bigwedge_{i=1}^{\ell} C_i \models D$

Motivation

- allow NP oracles to collapse arbitrary propositional derivations into one step
- akin to using SAT calls in QBF solving

Reasons for QBF hardness

NP oracles in QBF proof systems

- eliminate propositional hardness
- What sources of hardness exist for these QBF systems?

Answer

- circuit complexity lower bounds

Precise characterisations in QBF

Theorem [B., Bonacina, Chew & Pich 20]

There exist hard formulas in *Q-Frege* if and only if there exist

- lower bounds for propositional Frege or
- there exist lower bounds for non-uniform NC^1 (more precisely $\mathrm{PSPACE} \not\subseteq \mathrm{NC}^1$).

Alternative formulation

- super-polynomial lower bounds for $Q\text{-}Frege^{NP}$ iff $\mathrm{PSPACE} \not\subseteq \mathrm{NC}^1$
- super-polynomial lower bounds for $Q\text{-}EF^{NP}$ iff $\mathrm{PSPACE} \not\subseteq \mathrm{P/poly}$
- works for all the 'usual' Frege systems: AC^0, $AC^0[p]$, TC^0, NC^1, $P/poly$

Strategy extraction by decision lists

A $\mathcal{C}$-decision list computes a function $u = f(x_1, \ldots, x_n)$

IF $C_1(x_1, \ldots, x_n) = 1$ THEN $u \leftarrow b_1$
ELSE IF $C_2(x_1, \ldots, x_n) = 1$ THEN $u \leftarrow b_2$
$\vdots$
ELSE IF $C_\ell(x_1, \ldots, x_n) = 1$ THEN $u \leftarrow b_\ell$
ELSE $u \leftarrow c_{\ell+1}$ where $C_i \in \mathcal{C}$ and $b_i \in \{0, 1\}$

Theorem (B., Bonacina, Chew 15)

Q-$\mathcal{C}$-Frege has strategy extraction in $\mathcal{C}$-decision lists, i.e. from a refutation π of $F(\vec{x}, \vec{u})$ we can extract in poly-time a collection of $\mathcal{C}$-decision lists computing a winning strategy on the universal variables of F.

From decision lists to circuits

$\textsc{If } C_1(x_1, \ldots, x_n) = 1 \textsc{ Then } u \leftarrow b_1$
$\textsc{Else If } C_2(x_1, \ldots, x_n) = 1 \textsc{ Then } u \leftarrow b_2$
$\vdots$
$\textsc{Else If } C_\ell(x_1, \ldots, x_n) = 1 \textsc{ Then } u \leftarrow b_\ell$
$\textsc{Else } u \leftarrow c_{\ell+1}$ where $C_i \in \mathcal{C}$ and $b_i \in \{0, 1\}$

Proposition

Each $\mathcal{C}$-decision list as above can be transformed into a $\mathcal{C}$-circuit of depth $\max(\mathrm{depth}(C_i)) + 2$.

Corollary

- *Q-depth-d-Frege has strategy extraction with circuits of depth* $d + 2$.
- *Q-*AC^0*-Frege has strategy extraction in* AC^0.
- *Q-*$\mathsf{AC}^0[p]$*-Frege has strategy extraction in* $\mathsf{AC}^0[p]$.

From functions to QBFs

- Let $f(\vec{x})$ be a Boolean function.
- Define the QBF

$$\text{Q-}f = \exists\vec{x}\forall z\exists\vec{t}.\ z \neq f(\vec{x})$$

- $\vec{t}$ are auxiliary variables describing the computation of a circuit for f.
- $z \neq f(\vec{x})$ is encoded as a CNF.
- The only winning strategy for the universal player is to play $z \leftarrow f(\vec{x})$.

From circuit lower bounds to proof size lower bounds

Theorem (B., Bonacina, Chew 15)

Let f be any function hard for depth 3 circuits.
Then Q-f is hard for Q-Res.

Proof.

- Let Π be a refutation of Q-f in Q-Res.
- By strategy extraction, we obtain from Π a decision list computing f.
- Transform the decision list into a depth 3 circuit C for f.
- As f is hard to compute in depth 3, Π must be long.

□

Strong lower bound example I

Theorem (Razborov 1987, Smolensky 1987)

For each odd prime p, Parity requires exponential-size $\mathrm{AC}^0[p]$ *circuits.*

Corollary

Q-Parity requires exponential-size Q-$\mathrm{AC}^0[p]$*-Frege proofs.*

In contrast

no lower bound is known for $\mathrm{AC}^0[p]$-Frege.

Strong separations

Theorem (Smolensky 1987)

MOD_q requires exponential-size $\mathrm{AC}^0[p]$ circuits, where p and q are distinct primes.

Carefully choosing the formulas representing MOD_q we get:

Corollary (B., Bonacina, Chew 15)

For each pair p, q of distinct primes the MOD_q-formulas

- *require exponential-size proofs in Q-$\mathrm{AC}^0[p]$-Frege,*
- *but have polynomial-size proofs in Q-$\mathrm{AC}^0[q]$-Frege.*

Corollary

Q-$\mathrm{AC}^0[p]$-Frege is exponentially weaker than Q-TC^0-Frege.

In the propositional case

these separations are wide open.

Strong lower bound example II

Theorem (Håstad 1989)

The functions Sipser$_d$ exponentially separate depth $d-1$ from depth d circuits.

Theorem (B., Bonacina, Chew 2015)

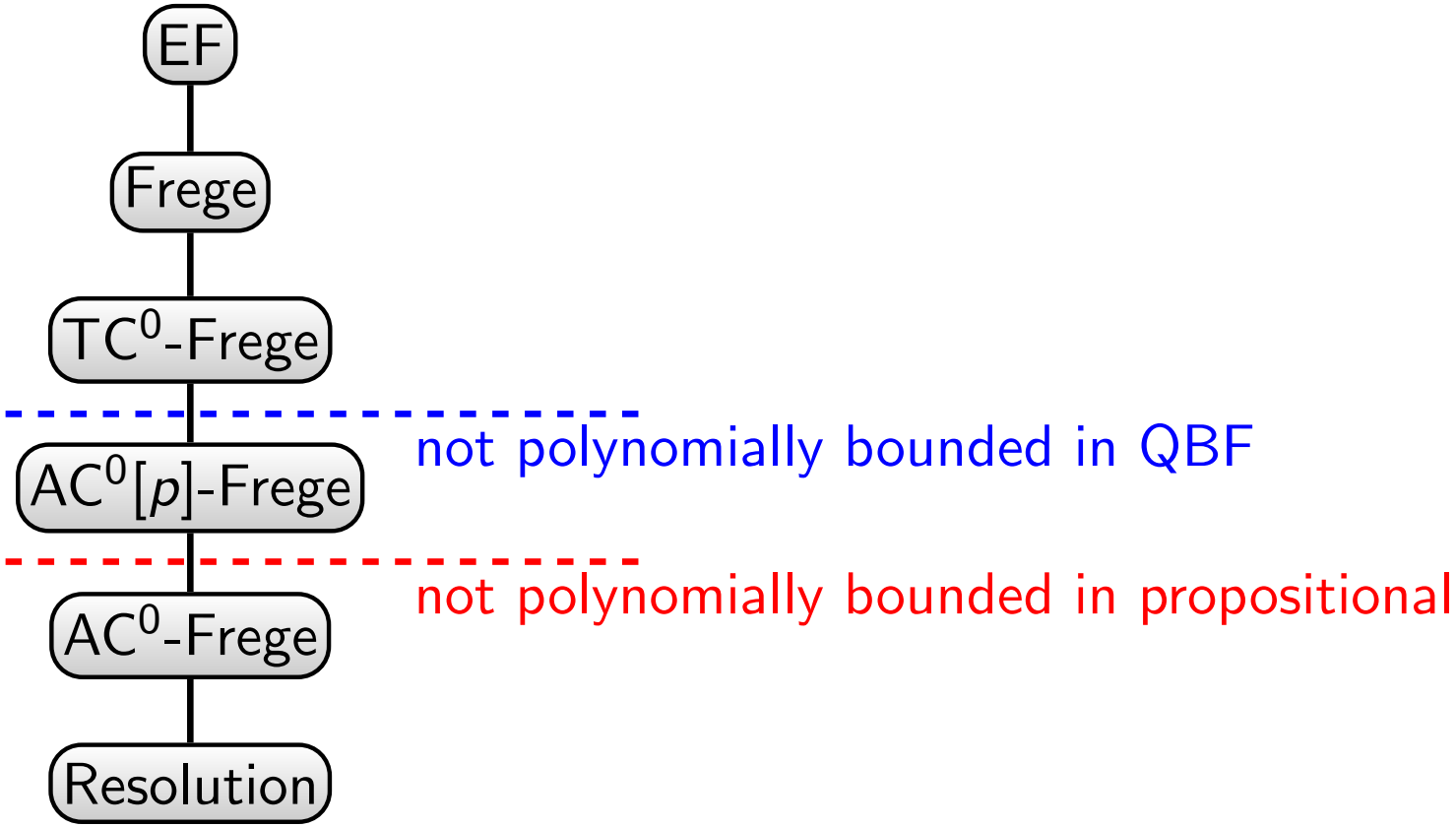
Q-Sipser$_d$

- *requires exponential-size proofs in depth $(d-3)$-Q-Frege.*
- *has polynomial-size proofs in depth d-Q-Frege.*

Note

- Q-Sipser$_d$ is a quantified CNF.
- Separating depth d Frege systems with constant depth formulas (independent of d) is a major open problem in the propositional case.

The current frontier: propositional vs QBF



What happens for resolution?

Question

- Can we characterise QBF resolution hardness by circuit complexity?
- QBF resolution corresponds to QBF solving.

Answer

- tight characterisation of QBF resolution by a decision list model
- as a 'by-product': size-width relation for QBF resolution

Unified decision lists

Our circuit model

- natural multi-output generalisation of decision lists [Rivest 87]
- computes functions $\{0,1\}^n \to \{0,1\}^m$
- input variables $x_1, \ldots, x_n$
- output variables $u_1, \ldots, u_m$

$$
\begin{aligned}
&\textsc{If } t_1 \textsc{ Then } \vec{u} = \vec{b}_1 \\
&\textsc{Else If } t_2 \textsc{ Then } \vec{u} = \vec{b}_2 \\
&\qquad\vdots \\
&\textsc{Else If } t_k \textsc{ Then } \vec{u} = \vec{b}_k \\
&\textsc{Else } \vec{u} = \vec{b}_{k+1}
\end{aligned}
$$

- t_i are terms in $x_1, \ldots, x_n$
- $\vec{b}_i$ are total assignments to $u_1, \ldots, u_m$

We call this model unified decision lists (UDL).

The characterisation for Q-Res

Theorem [B., Blinkhorn, Mahajan 20]

- Let Φ be a false QBF of bounded quantifier complexity.
- Then the size of the smallest $Q\text{-}Res^{NP}$ refutation of Φ is polynomially related to the size of the smallest UDL for Φ.

Equivalently

A sequence Φ_n of bounded quantification is hard for $Q\text{-}Res$ if and only if

1. Φ_n require large UDLs, or
2. Φ_n contain propositional resolution hardness.

Remark

The propositional resolution hardness in 2. can be precisely identified.

Comparison to QBF Frege

In QBF Frege

- hardness in $Q\text{-}Frege^{NP}$ working with lines from $\mathcal{C}$ is characterised precisely by hardness for $\mathcal{C}$ circuits [B. & Pich 16].

In QBF resolution

- we work with CNFs (depth-2 circuits).
- Complexity of decision lists (and hence UDLs) is strictly intermediate between depth-2 and depth-3 circuits [Krause 06].
- Hence, circuit characterisation of QBF resolution by a slightly stronger model than used in the proof system.

Proof ingredients – Part 1

From proofs to circuits

- From a $Q\text{-}Res^{NP}$ efficiently extract a winning strategy for the universal player in terms of a UDL.
- Strategy extraction for each universal variable previously known via single-output decision lists [Balabanov & Jiang 12],[B., Bonacina & Chew 16]
- Need to be combined into one UDL (this step depends on quantifier complexity).

Remarks

- Single output decision lists provably too strong to characterise $Q\text{-}Res^{NP}$ hardness.
- There exist QBFs hard for $Q\text{-}Res^{NP}$, but with trivial single-output decision lists.

Proof ingredients – Part 2

From circuits to proofs

- We construct a normal form for a $Q\text{-}Res^{NP}$ refutation of Φ via an entailment sequence from a UDL for Φ.
- Intuition: entailment sequence proves the correctness of the UDL.
- Entailment sequence allows to identify propositional resolution hardness.

Conclusion

- Tight correspondence between QBF proof systems and circuit classes
- works for QBF Frege systems and QBF resolution (bounded quantifier complexity)
- allows to elegantly prove many lower bounds

Open problems

- find the right circuit models for
  - resolution with unbounded QBFs (UDLs too weak)
  - QBF resolutions systems corresponding to QBF solving
  - further systems: QBF cutting planes, ...