Simplified and Improved Separations Between Regular and General Resolution by Lifting

Marc Vinyals

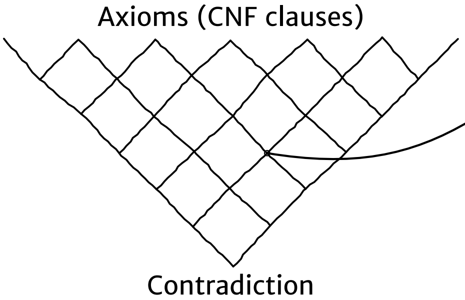
Technion
Haifa, Israel

joint work with Jan Elffers, Jan Johannsen, and Jakob Nordström

Background

Regular Resolution

'37 **Resolution.** [Blake] **Res**

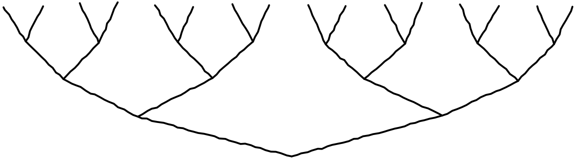


$$\frac{C \vee x \quad D \vee \overline{x}}{C \vee D}$$

Regular Resolution

'37 Resolution. [Blake]

'62 Tree-like resolution. [DPLL]



Res

Tree Res

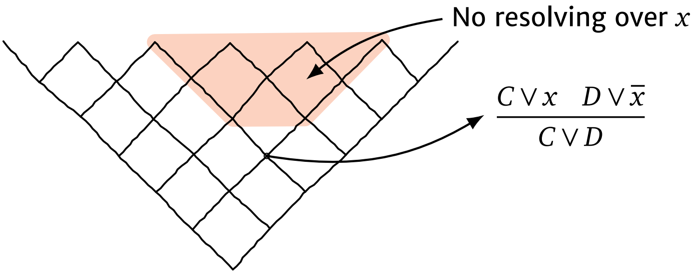
Regular Resolution

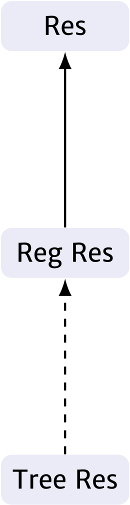
'37 Resolution. [Blake]

'62 Tree-like resolution. [DPLL]

'68 Regular resolution: do not resolve a variable twice on same path. [Tseitin]

- ► Tree-like resolution is regular wlog.
- Q Is regular resolution as powerful as general resolution?





Regular Resolution

'37 Resolution. [Blake]

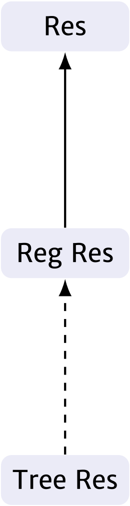
'62 Tree-like resolution. [DPLL]

'68 Regular resolution: do not resolve a variable twice on same path. [Tseitin]

- Tree-like resolution is regular wlog.

Q Is regular resolution as powerful as general resolution?

- Formulas need exponentially long regular proofs. [Tseitin,Galil]
- If regular $\equiv$ general, resolution needs exponentially long proofs.



Regular Resolution

’37 Resolution. [Blake]

’62 Tree-like resolution. [DPLL]

’68 Regular resolution: do not resolve a variable twice on same path. [Tseitin]

- Tree-like resolution is regular wlog.

Q Is regular resolution as powerful as general resolution?

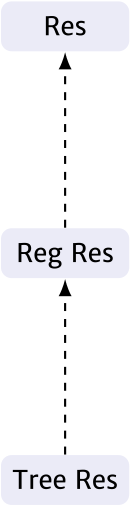
- Formulas need exponentially long regular proofs. [Tseitin,Galil]
- If regular $\equiv$ general, resolution needs exponentially long proofs.

’87 Separation regular vs general (by a constant). [Huang, Yu]

’93 Separation regular vs general (superpolynomial). [Goerdt]

’02 Separation regular vs general (exponential). [AJPU]

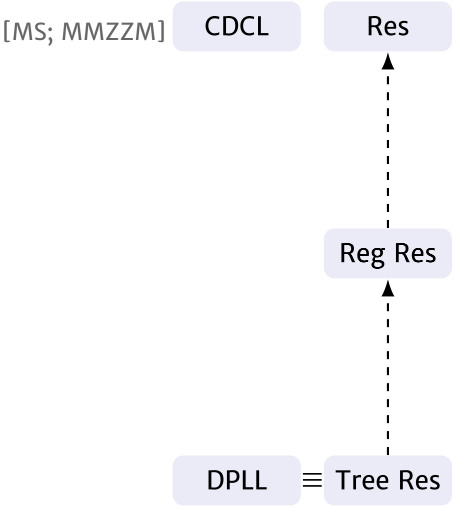
’11 Best separation to date: $\exp(L/\log^7 L \log\log L)$. [Urquhart]



CDCL and Restarts

'96 CDCL: DPLL + Learning

- Also: VSIDS, Restarts.



CDCL and Restarts

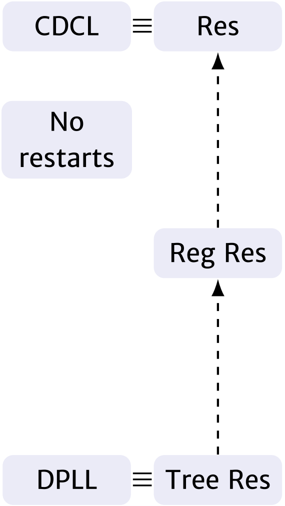
’96 CDCL: DPLL + Learning [MS; MMZZM]

- Also: VSIDS, Restarts.

’09 CDCL as powerful as resolution. [PD; AFT]

- Crucially uses restarts.
- Restarts also seem very important in practice.

Q Are restarts really needed?



CDCL and Restarts

'96 CDCL: DPLL + Learning [MS; MMZZM]

- Also: VSIDS, Restarts.

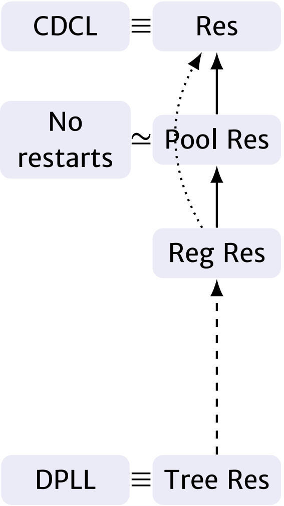
'09 CDCL as powerful as resolution. [PD; AFT]

- Crucially uses restarts.
- Restarts also seem very important in practice.

Q Are restarts really needed?

'05 Pool resolution $\simeq$ CDCL w/o restarts. [van Gelder]

- Pool res $\geq$ Regular res $\Rightarrow$ Formulas that separate general and regular are good candidates to separate general and pool.



CDCL and Restarts

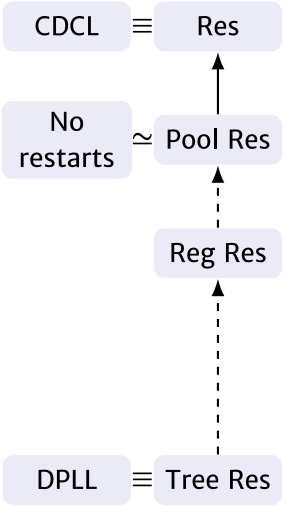
'96 CDCL: DPLL + Learning [MS; MMZZM]
- Also: VSIDS, Restarts.

'09 CDCL as powerful as resolution. [PD; AFT]
- Crucially uses restarts.
- Restarts also seem very important in practice.

Q Are restarts really needed?

'05 Pool resolution $\simeq$ CDCL w/o restarts. [van Gelder]
- Pool res $\geq$ Regular res $\Rightarrow$ Formulas that separate general and regular are good candidates to separate general and pool.

'14 All such formulas easy for pool resolution. [BBJ; BK]
- Also: formulas not good to run experiments with.
- Need new formulas!



Proving Resolution Lower Bounds

Largest clause in proof

Size–Width Relation

Resolution	F requires width $W \Rightarrow F$ requires length $\exp(W^2/n)$
Tree–like resolution	F requires width $W \Rightarrow F$ requires length $\exp(W)$
Regular resolution	??

Proving Resolution Lower Bounds

Largest clause in proof

Size–Width Relation

Resolution	F requires width $W \Rightarrow F$ requires length $\exp(W^2/n)$
Tree-like resolution	F requires width $W \Rightarrow F$ requires length $\exp(W)$
Regular resolution	??

Lifting

Resolution	F requires width $W \Rightarrow T(F)$ requires length $\exp(W)$
Tree-like resolution	F requires depth $D \Rightarrow T(F)$ requires length $\exp(D)$
Regular resolution	??

Longest path in proof DAG

Results

Main Result (Informal)

Theorem

F requires large depth $\Rightarrow T(F)$ requires long regular proofs.

Main Result (Informal)

Theorem

F requires large depth $\Rightarrow T(F)$ requires long regular proofs.

- Simplifies separation between regular and general resolution.
  - If F has narrow proofs, then $T(F)$ still has short proofs.
  - Obtain separation from F with small width and large depth, e.g. pebbling formulas.

Main Result (Informal)

Theorem

F requires large depth $\Rightarrow T(F)$ requires long regular proofs.

- Simplifies separation between regular and general resolution.
  - If F has narrow proofs, then $T(F)$ still has short proofs.
  - Obtain separation from F with small width and large depth, e.g. pebbling formulas.

- New family of "sparse stone formulas".
- Improved separation: $\exp(L/\log^3 L \log\log^5 L)$.
- Can use in experiments.

Lifting

Usual Lifting

- Replace each original variable x_i with a gadget $g_i(y_i^1, \ldots, y_i^k)$.
- e.g. $x_1 \lor \neg x_2 \quad \to \quad (y_1^1 \oplus y_1^2) \lor \neg(y_2^1 \oplus y_2^2)$.

Lifting

Usual Lifting

- Replace each original variable x_i with a gadget $g_i(y_i^1, \ldots, y_i^k)$.
- e.g. $x_1 \lor \neg x_2 \quad \rightarrow \quad (y_1^1 \oplus y_1^2) \lor \neg(y_2^1 \oplus y_2^2)$.

Lifting with Reusing

- Share variables among gadgets.

Lifting

Selector variables

Lifting with Indexing

Main variables

- Gadget $g_i(s_i^1, \ldots, s_i^m; r_i^1, \ldots, r_i^m)$: if $s_i^j = 1$, then $g_i(\cdots) = r_i^j$.
  (Assume exactly one s_i variable is 1)

Lifting

Selector variables

Main variables

Lifting with Indexing

- Gadget $g_i(s_i^1, \ldots, s_i^m; r_i^1, \ldots, r_i^m)$: if $s_i^j = 1$, then $g_i(\cdots) = r_i^j$.
  (Assume exactly one s_i variable is 1)

Lifting with Indexing and Reusing

- Share all main variables among all gadgets.

Lifting

Selector variables

Main variables

Lifting with Indexing

- Gadget $g_i(s_i^1,\ldots,s_i^m;r_i^1,\ldots,r_i^m)$: if $s_i^j=1$, then $g_i(\cdots)=r_i^j$.
  (Assume exactly one s_i variable is 1)

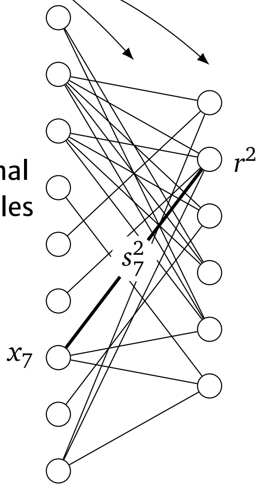
Lifting with Indexing and Reusing

- Share all main variables among all gadgets.

Lifting with Sparse Indexing and Reusing

- Fix a bipartite graph $G([n]\cup[m],E)$; variable s_i^j exists iff $(i,j)\in E$.
- G is n disjoint stars $\Rightarrow$ usual lifting.
- F is pebbling formula and G is complete graph $K_{n,m}$ $\Rightarrow$ stone formula.
- F is pebbling formula and G is random graph $\Rightarrow$ sparse stone formula.

Original variables



Main Result

Theorem (Dense)

If F requires depth D, then $\mathcal{L}_K(F)$ requires regular length $\sim \exp(D^2/n)$.

Main Result

Theorem (Dense)

If F requires depth D, then $\mathcal{L}_K(F)$ requires regular length $\sim \exp(D^2/n)$.

Theorem (Sparse)

If F requires depth D, then $\mathcal{L}_G(F)$ requires regular length $\sim \exp(D^3/n^2 \log^2 n)$.

G is a random graph of degree $d = \log(n/D)$.

Proof

Proof Overview (Dense)

Random restriction technique

1. Hit proof with random restriction ρ.
2. If proof of F is short, obtain proof of $F\restriction_\rho = F'$ with no wide clauses.
3. But all proofs of F' have a wide clause.

Proof Overview (Dense)

Random restriction technique

1. Hit proof with random restriction ρ.
2. If proof of F is short, obtain proof of $F\restriction_\rho = F'$ with no wide clauses.
3. But all proofs of F' have a wide clause.

- Need restriction to respect lifting: $\mathcal{L}(F)\restriction_\rho = F' = \mathcal{L}(F'')$. [AJPU '02]
- Need to tweak what "wide" means.
- Clause is "complex" if [AJPU '02]
  - talks about many main variables or
  - matches many original variables or
  - restricts the neighbourhood of many original variables

Proof Overview (Dense)

Updated plan

1. Hit proof with lifting-respecting restriction ρ.
2. If proof of $\mathcal{L}(F)$ is short, obtain proof of $\mathcal{L}(F)\restriction_\rho = \mathcal{L}(F'')$ with no complex clauses.
3. But all proofs of $\mathcal{L}(F'')$ have a complex clause.

- Need restriction to respect lifting: $\mathcal{L}(F)\restriction_\rho = F' = \mathcal{L}(F'')$. [AJPU '02]
- Need to tweak what "wide" means.
- Clause is "complex" if [AJPU '02]
  - talks about many main variables or
  - matches many original variables or
  - restricts the neighbourhood of many original variables

Proof Details (Dense)

- Finding a good restriction is not too hard.

Proof Details (Dense)

Can query and forget
but not requery

Given truth assignment α,
find clause falsified by α

- Finding a good restriction is not too hard.
- To prove that all **regular** proofs have a complex clause:
  - view proof as **read-once** branching program for $\mathrm{Search}(\mathcal{L}(F))$
  - use to build decision tree for $\mathrm{Search}(F)$.

Proof Details (Dense)

Can query and forget
but not requery

Given truth assignment α,
find clause falsified by α

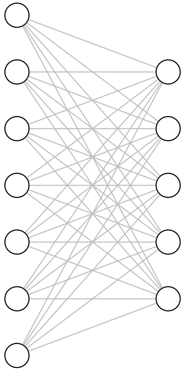
- Finding a good restriction is not too hard.
- To prove that all **regular** proofs have a complex clause:
  - view proof as **read-once** branching program for $\mathrm{Search}(\mathcal{L}(F))$
  - use to build decision tree for $\mathrm{Search}(F)$.
- Key invariant: match original variables to main variables consistent with decision tree.
  - If query selector variable: say "not matched" unless forced to.
  - If query main variable: if matched, answer according to decision tree.

Proof Details (Dense)

Can query and forget but not requery

Given truth assignment α, find clause falsified by α

- Finding a good restriction is not too hard.
- To prove that all **regular** proofs have a complex clause:
  - view proof as **read-once** branching program for $\mathrm{Search}(\mathcal{L}(F))$
  - use to build decision tree for $\mathrm{Search}(F)$.
- Key invariant: match original variables to main variables consistent with decision tree.
  - If query selector variable: say "not matched" unless forced to.
  - If query main variable: if matched, answer according to decision tree.



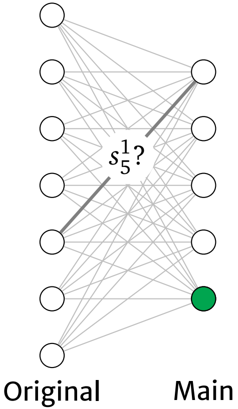
Proof Details (Dense)

Can query and forget but not requery

- Finding a good restriction is not too hard.

Given truth assignment α, find clause falsified by α

- To prove that all **regular** proofs have a complex clause:
  - view proof as **read-once** branching program for $\mathrm{Search}(\mathcal{L}(F))$
  - use to build decision tree for $\mathrm{Search}(F)$.



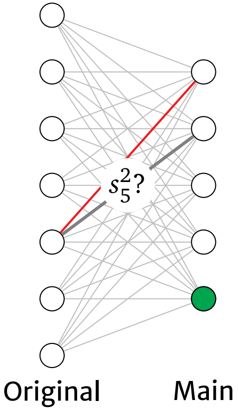
- Key invariant: match original variables to main variables consistent with decision tree.
  - If query selector variable: say "not matched" unless forced to.
  - If query main variable: if matched, answer according to decision tree.

Proof Details (Dense)

Can query and forget
but not requery

Given truth assignment α,
find clause falsified by α

- Finding a good restriction is not too hard.
- To prove that all **regular** proofs have a complex clause:
  - view proof as **read-once** branching program for $\mathrm{Search}(\mathcal{L}(F))$
  - use to build decision tree for $\mathrm{Search}(F)$.
- Key invariant: match original variables to main variables consistent with decision tree.
  - If query selector variable: say "not matched" unless forced to.
  - If query main variable: if matched, answer according to decision tree.



Proof Details (Dense)

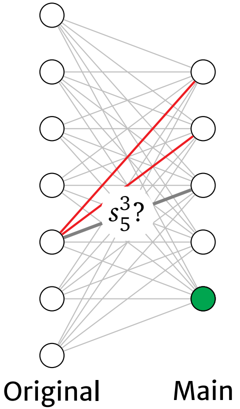
Can query and forget
but not requery

- Finding a good restriction is not too hard.

Given truth assignment α,
find clause falsified by α

- To prove that all **regular** proofs have a complex clause:
  - view proof as **read-once** branching program for $\mathrm{Search}(\mathcal{L}(F))$
  - use to build decision tree for $\mathrm{Search}(F)$.

- Key invariant: match original variables to main variables consistent with decision tree.
  - If query selector variable: say "not matched" unless forced to.
  - If query main variable: if matched, answer according to decision tree.



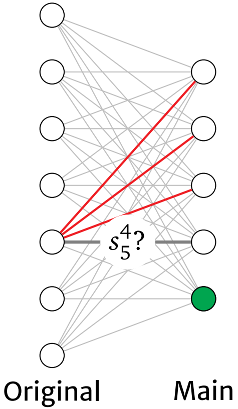
Proof Details (Dense)

Can query and forget
but not requery

- Finding a good restriction is not too hard.

Given truth assignment α,
find clause falsified by α

- To prove that all **regular** proofs have a complex clause:
  - view proof as **read-once** branching program for $\mathrm{Search}(\mathcal{L}(F))$
  - use to build decision tree for $\mathrm{Search}(F)$.



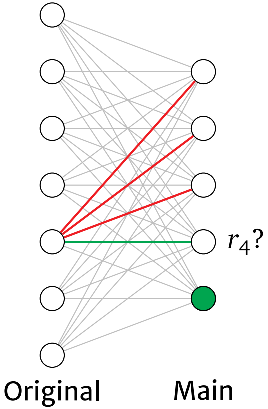
- Key invariant: match original variables to main variables consistent with decision tree.
  - If query selector variable: say "not matched" unless forced to.
  - If query main variable: if matched, answer according to decision tree.

Proof Details (Dense)

Can query and forget but not requery

Given truth assignment α, find clause falsified by α

- Finding a good restriction is not too hard.
- To prove that all **regular** proofs have a complex clause:
  - view proof as **read-once** branching program for $\mathrm{Search}(\mathcal{L}(F))$
  - use to build decision tree for $\mathrm{Search}(F)$.
- Key invariant: match original variables to main variables consistent with decision tree.
  - If query selector variable: say "not matched" unless forced to.
  - If query main variable: if matched, answer according to decision tree.

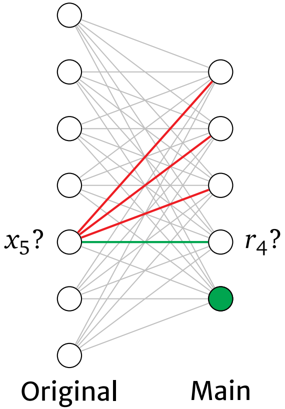


Proof Details (Dense)

Can query and forget but not requery

Given truth assignment α, find clause falsified by α

- Finding a good restriction is not too hard.
- To prove that all **regular** proofs have a complex clause:
  - view proof as **read-once** branching program for $\mathrm{Search}(\mathcal{L}(F))$
  - use to build decision tree for $\mathrm{Search}(F)$.
- Key invariant: match original variables to main variables consistent with decision tree.
  - If query selector variable: say "not matched" unless forced to.
  - If query main variable: if matched, answer according to decision tree.

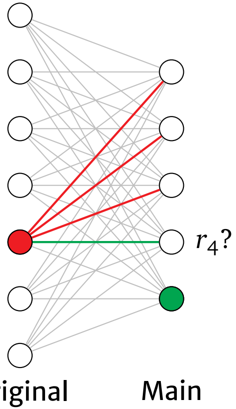


Proof Details (Dense)

Can query and forget but not requery

Given truth assignment α, find clause falsified by α

- Finding a good restriction is not too hard.
- To prove that all **regular** proofs have a complex clause:
  - view proof as **read-once** branching program for $\mathrm{Search}(\mathcal{L}(F))$
  - use to build decision tree for $\mathrm{Search}(F)$.
- Key invariant: match original variables to main variables consistent with decision tree.
  - If query selector variable: say "not matched" unless forced to.
  - If query main variable: if matched, answer according to decision tree.



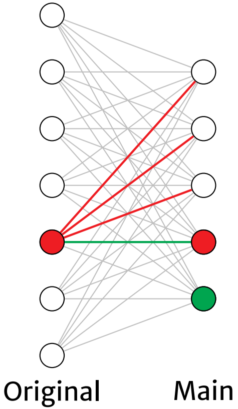
Proof Details (Dense)

Can query and forget
but not requery

- Finding a good restriction is not too hard.

Given truth assignment α,
find clause falsified by α

- To prove that all **regular** proofs have a complex clause:
  - view proof as **read-once** branching program for $\mathrm{Search}(\mathcal{L}(F))$
  - use to build decision tree for $\mathrm{Search}(F)$.
- Key invariant: match original variables to main variables consistent with decision tree.
  - If query selector variable: say "not matched" unless forced to.
  - If query main variable: if matched, answer according to decision tree.
- If no complex clause, then a coloured main variable is never matched
  - Hence must query D main variables.
  - Hence (read once) must query D different main variables.
  - Contradiction, only have $m < D$ main variables.



Experiments

Experiments

- Experiments with sparse stone formulas.

In theory...

- Short proofs always exist.
- 100s variables, $10\,000$s clauses.

Experiments

- Experiments with sparse stone formulas.

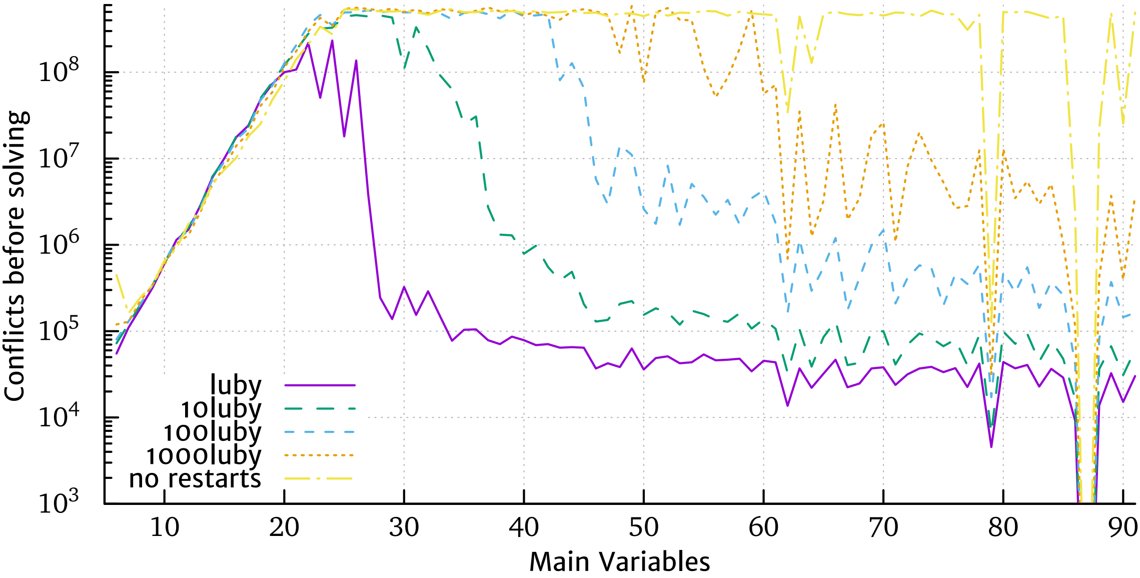
In theory...

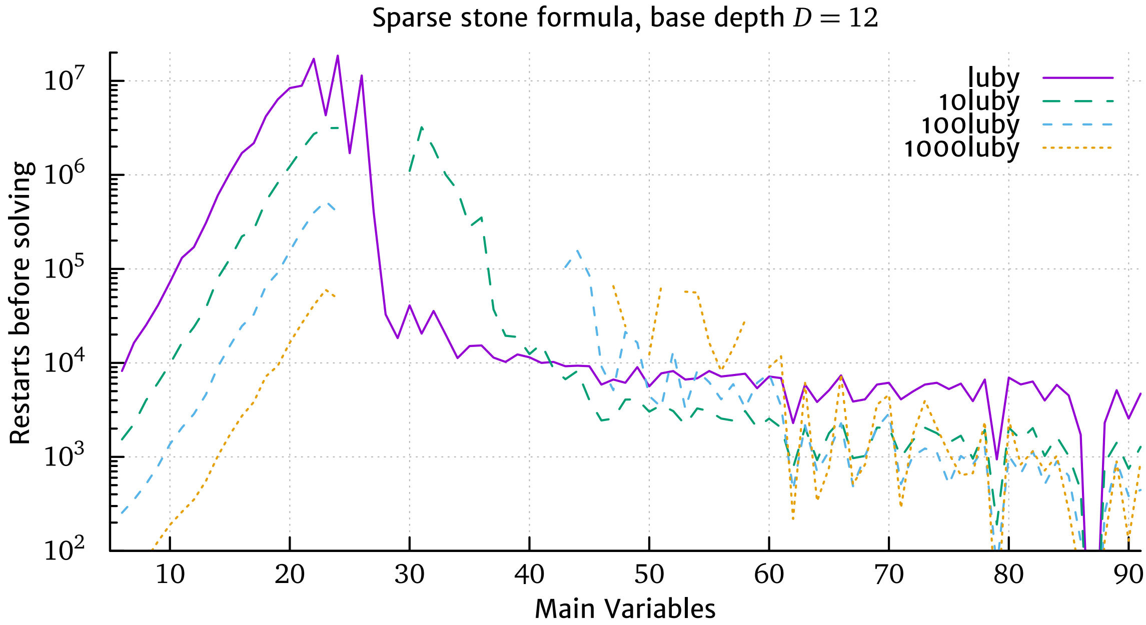
- Short proofs always exist.
- 100s variables, 10 000s clauses.

In practice...

- Few main variables $\Rightarrow$ very hard.
- Many main variables $\Rightarrow$ restarts crucial.

Sparse stone formula, base depth $D = 12$





Sparse stone formula, base depth $D = 12$
Restarts before solving
Main Variables
10^2
10^3
10^4
10^5
10^6
10^7
10
20
30
40
50
60
70
80
90
luby
10luby
100luby
1000luby

Take Home

Results

- Generic tool to prove regular resolution lower bounds: lifting with reusing.
- New and simplified lower bounds for regular resolution.

Take Home

Results

- Generic tool to prove regular resolution lower bounds: lifting with reusing.
- New and simplified lower bounds for regular resolution.

Open Problems

- Are restarts needed?
- More formulas that separate regular and general resolution?

Take Home

Results

- Generic tool to prove regular resolution lower bounds: lifting with reusing.
- New and simplified lower bounds for regular resolution.

Open Problems

- Are restarts needed?
- More formulas that separate regular and general resolution?

Thanks!