Partial Regularization of First-Order Resolution Proofs

J. Gorzny[1] B. Woltzenlogel Paleo[2,3]

[1]University of Waterloo

[2]Australian National University

[3]Vienna University of Technology

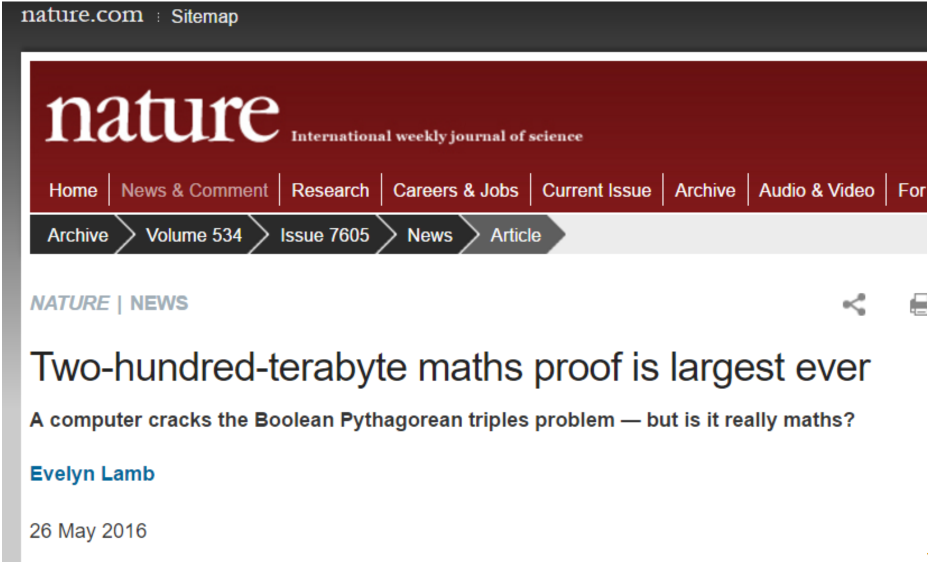
23 November 2016

The Quest for Simple Proofs

"The 24th problem in my Paris lecture was to be: Criteria of simplicity, or proof of the greatest simplicity of certain proofs. Develop a theory of the method of proof in mathematics in general. Under a given set of conditions there can be but one simplest proof. Quite generally, if there are two proofs for a theorem, you must keep going until you have derived each from the other, or until it becomes quite evident what variant conditions (and aids) have been used in the two proofs. "

–David Hilbert [Thi03]

The 'Real World'


nature.com : Sitemap

nature International weekly journal of science

Home | News & Comment | Research | Careers & Jobs | Current Issue | Archive | Audio & Video | For

Archive > Volume 534 > Issue 7605 > News > Article

NATURE | NEWS

Two-hundred-terabyte maths proof is largest ever

A computer cracks the Boolean Pythagorean triples problem — but is it really maths?

Evelyn Lamb

26 May 2016

(See [HKM16])

First-Order Proof Compression Motivation

- The best, most efficient provers, do not generate the best, least redundant proofs.
- Many compression algorithms for propositional proofs; few for first-order proofs.
- Finding a minimal proof is NP-hard, so use heuristics to find smaller proofs (see [FMP11])

Our Goal

Lifting propositional proof compression algorithms to first-order logic.

This work: `LowerUnits` [FMP11] and `RecyclePivotWithIntersection` [FMP11, BIFH$^+$08]

(Propositional) Proofs

Definition (Proof)

A directed acyclic graph $\langle V, E, \Gamma \rangle$, where

- V is a set of nodes
- E is a set of edges labeled by literals
- Γ (the proof clause) is inductively constructible using *axiom* and *resolution* nodes

Definition (Axiom)

A proof with a single node (so $E = \emptyset$)

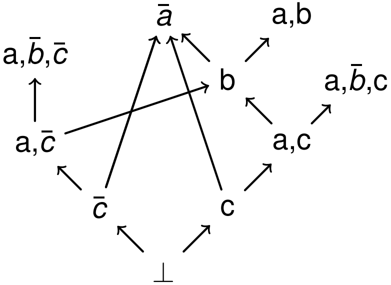
(Propositional) Resolution

Definition (Resolution)

Given two proofs ψ_L and ψ_R with conclusions Γ_L and Γ_R with some literal l such that $\bar{l} \in \Gamma_L$ and $l \in \Gamma_R$, the resolution proof ψ of ψ_L and ψ_R on l, denoted $\psi = \psi_L \psi_R$ is such that:

- ψ's nodes are the union of the nodes of ψ_L and ψ_R, and a new root node
- there is an edge from $\rho(\psi)$ to $\rho(\psi_L)$ labeled with $\bar{l}$
- there is an edge from $\rho(\psi)$ to $\rho(\psi_R)$ labeled with l
- ψ's conclusion is $(\Gamma_L \setminus \{\bar{l}\}) \cup (\Gamma_R \setminus \{l\})$

A Propositional Proof



Deletion

Deletion of an edge

- The resolvent is replaced by the other premise
- Some subsequent resolutions may have to be deleted too

Deletion of a subproof ψ

- Deletion of every edge coming to $\rho(\psi)$
- The operation is commutative and associative

First-Order Proofs

Definition (First-Order Proof)

A directed acyclic graph $\langle V, E, \Gamma \rangle$, where

- V is a set of nodes
- E is a set of edges labeled by literals **and substitutions**
- Γ (the proof clause) is inductively constructible using *axiom*, ***(first-order) resolution***, **and *contraction*** nodes

Axioms are unchanged

Substitutions and Unifiers

Definition (Substitution)

A mapping $\{X_1 \setminus t_1, X_2 \setminus t_2, \ldots\}$ from variables $X_1, X_2, \ldots$ to terms $t_1, t_2, \ldots$

Definition (Unifier)

A substitution that makes two terms equal when applied to them.

First-Order (Unifying) Resolution

Definition (First-Order Resolution)

Given two proofs ψ_L and ψ_R with conclusions Γ_L and Γ_R with some literal l such that $l_L \in \Gamma_L$ and $l_R \in \Gamma_R$, and σ_L and σ_R are substitutions usch that $l_L\sigma_L = \overline{l_R}\sigma_R$, and the variables in $(\Gamma_L \setminus l_L)\sigma_L$ and $(\Gamma_R \setminus l_R)\sigma_R$ are disjoint, then the resolution proof ψ of ψ_L and ψ_R on l, denoted $\psi = \psi_L\psi_R$ is such that:

- ψ's nodes are the union of the nodes of ψ_L and ψ_R, and a new root node
- there is an edge from $\rho(\psi)$ to $\rho(\psi_L)$ labeled with l_L and σ_L
- there is an edge from $\rho(\psi)$ to $\rho(\psi_R)$ labeled with l_R and σ_R
- ψ's conclusion is $(\Gamma_L \setminus l_L)\sigma_L \cup (\Gamma_R \setminus l_R)\sigma_R$

Unifying Resolution Example

$$
\begin{tikzcd}
\eta_1\colon p(a) \vdash & & \eta_2\colon q(Y,X) \vdash p(Y) \\
& \psi\colon q(a,X) \vdash \arrow[ul] \arrow[ur] &
\end{tikzcd}
$$

$$\sigma = \{Y \rightarrow a\}$$

Refutation when $\psi = \bot$

Contraction

Definition (Contraction)

If ψ' is a proof and σ is a unifier of $\{l_1, \ldots, l_n\} \subset \Gamma'$, then a contraction ψ is a proof where

- ψ's nodes are the union of the nodes of ψ' and a new node v
- There is an edge from $\rho(\psi')$ to v labeled with $\{l_1, \ldots, l_n\}$ and σ
- The conclusion is $(\Gamma' \setminus \{l_1, \ldots, l_n\})\sigma \cup \{l\}$, where $l = l_k\sigma$ for $k \in \{1, \ldots, n\}$

Contraction Example

$$\eta_1\colon p(X, Y), p(a, Z), p(a, f(b)) \vdash q(Z)$$

$$\uparrow$$

$$\psi\colon p(a, f(b)) \vdash q(f(b))$$

$$\sigma = \{X \to a, Y \to f(b), Z \to f(b)\}$$

Contraction Example

$$\eta_1\colon p(X, Y), p(X, Z), p(U, V) \vdash q(Z)$$

$$\uparrow$$

$$\psi\colon p(X, Z) \vdash q(Z)$$

$$\sigma = \{Y \to Z, U \to Z, V \to Z\}$$

Contraction Example

$$\eta_1\colon p(X, Y), p(a, Z), p(a, f(b)) \vdash q(Z)$$

$$\psi\colon p(X, Y), p(a, f(b)) \vdash q(f(b))$$

$$\sigma = \{Z \rightarrow f(b)\}$$

Lowering Units

Definition (Unit)

A unit clause is a subproof with a conclusion clause (final clause) having exactly 1 literal

Theorem ([FMP11])

A unit clause can always be lowered

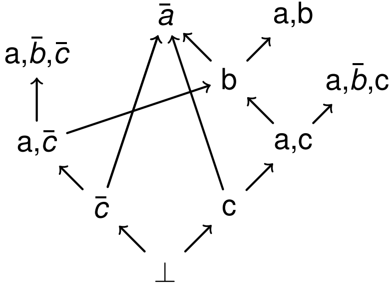
Compression is achieved by delaying resolution with unit clause subproofs.

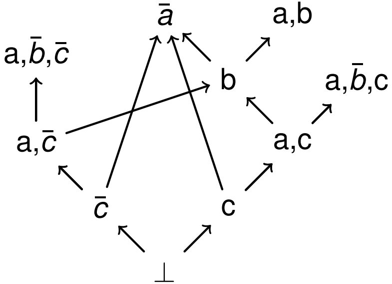
Two Traversals

LowerUnits

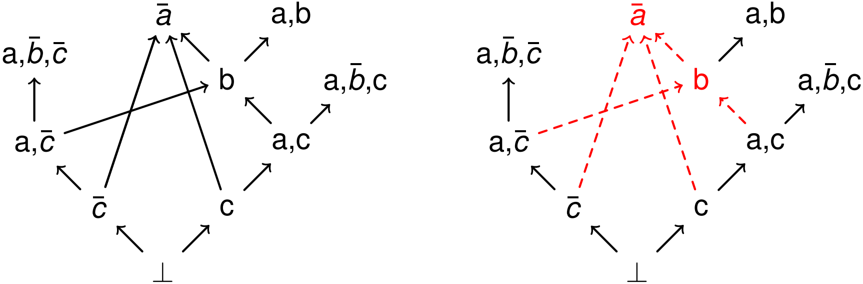
- $\uparrow$ Collect units with more than one resolvent
- $\downarrow$ Delete units and reintroduce them at the bottom of the proof

Propositional Example

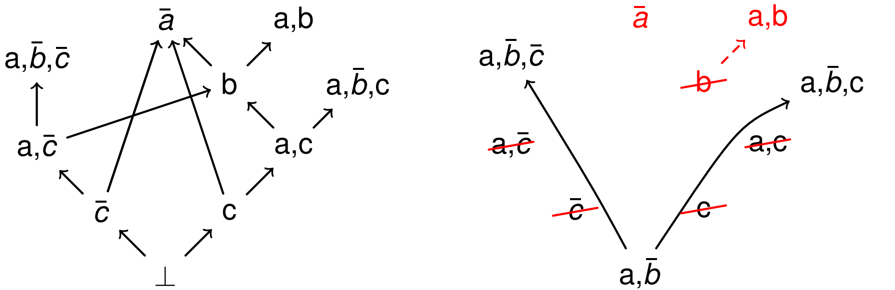




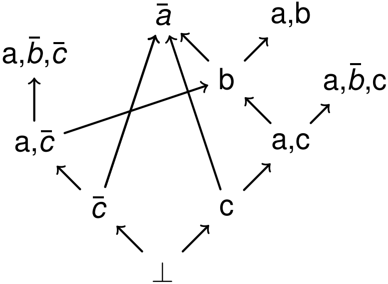
Propositional Example

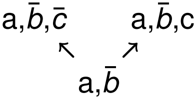


Propositional Example

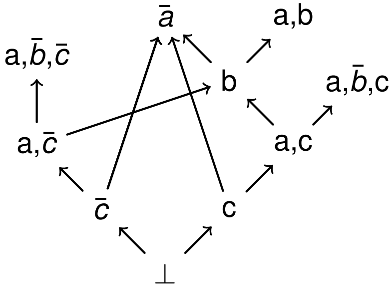


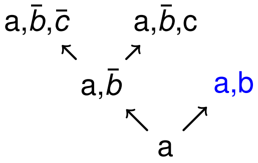
Propositional Example



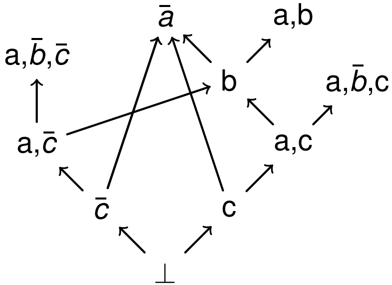


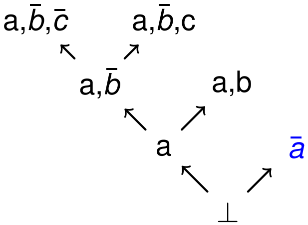
Propositional Example



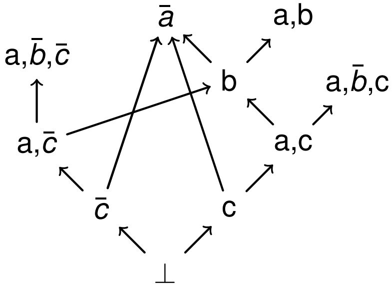


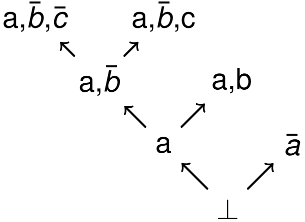
Propositional Example



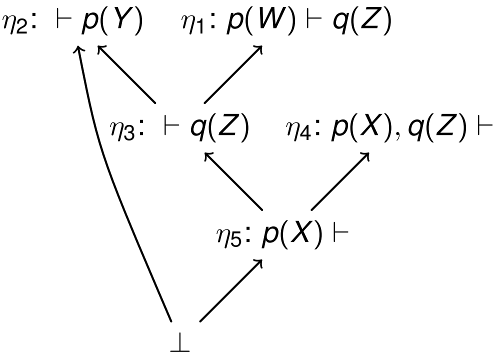


Propositional Example





First-Order Change: Helpful Contractions



First-Order Change: Helpful Contractions

$$
\begin{tikzcd}
\eta_2\colon \vdash p(Y) & & \eta_1\colon p(W) \vdash q(Z) & & \\
& \eta_3\colon \vdash q(Z) \arrow[ul] \arrow[ur] & & \eta_4\colon p(X), q(Z) \vdash & \\
& & \eta_5\colon p(X) \vdash \arrow[ul] \arrow[ur] & & \\
& \bot \arrow[uuul, bend left] \arrow[ur] & & &
\end{tikzcd}
$$

First-Order Change: Helpful Contractions

$$
\begin{tikzcd}[column sep=small]
{\color{red}\eta_2\colon \vdash p(Y)} & & \eta_1\colon p(W) \vdash q(Z) & \\
& \eta_3\colon \vdash q(Z) \arrow[ul] \arrow[ur] & & \eta_4\colon p(X), q(Z) \vdash \\
& & \eta_5\colon p(X) \vdash \arrow[ul] \arrow[ur] & \\
& \bot \arrow[uuul, bend left] \arrow[ur] & &
\end{tikzcd}
\qquad
\begin{tikzcd}[column sep=small]
\eta'_1\colon p(W) \vdash q(Z) & & \eta'_4\colon p(X), q(Z) \vdash \\
& \eta'_5 : p(X), p(Y) \vdash \arrow[ul] \arrow[ur] &
\end{tikzcd}
$$

First-Order Change: Helpful Contractions

$$
\begin{tikzcd}
\eta_2\colon \vdash p(Y) & & \eta_1\colon p(W) \vdash q(Z) & \\
& \eta_3\colon \vdash q(Z) \arrow[ul] \arrow[ur] & & \eta_4\colon p(X), q(Z) \vdash \\
& & \eta_5\colon p(X) \vdash \arrow[ul] \arrow[ur] & \\
& \bot \arrow[uuul] \arrow[ur] & &
\end{tikzcd}
\qquad
\begin{tikzcd}
\eta'_1\colon p(W) \vdash q(Z) & & \eta'_4\colon p(X), q(Z) \vdash \\
& \eta'_5 : p(X), p(Y) \vdash \arrow[ul] \arrow[ur] & \\
\lfloor \eta'_5 \rfloor\colon p(U) \vdash \arrow[ur] & &
\end{tikzcd}
$$

First-Order Change: Helpful Contractions

$$
\begin{tikzcd}
{\color{red}\eta_2\colon \vdash p(Y)} & & \eta_1\colon p(W) \vdash q(Z) & \\
& \eta_3\colon \vdash q(Z) \arrow[ul] \arrow[ur] & & \eta_4\colon p(X), q(Z) \vdash \\
& & \eta_5\colon p(X) \vdash \arrow[ul] \arrow[ur] & \\
& \bot \arrow[ur] \arrow[uuul, bend left] & &
\end{tikzcd}
\qquad
\begin{tikzcd}
\eta_1'\colon p(W) \vdash q(Z) & & \eta_4'\colon p(X), q(Z) \vdash \\
& \eta_5'\colon p(X), p(Y) \vdash \arrow[ul] \arrow[ur] & \\
\lfloor \eta_5' \rfloor\colon p(U) \vdash \arrow[ur] & & \eta_2'\colon \vdash p(Y) \\
& \bot \arrow[ul] \arrow[ur] &
\end{tikzcd}
$$

First-Order Challenge: Pre-Deletion Check

$$\begin{tikzcd}
\eta_5\colon q(Y) \vdash p(a) & & \eta_4\colon p(X) \vdash & & \eta_3\colon \vdash p(b), q(Y) \\
& \eta_1\colon q(Y) \vdash \arrow[ul] \arrow[ur] & & \eta_2\colon \vdash q(Y) \arrow[ul] \arrow[ur] & \\
& & \bot \arrow[ul] \arrow[ur] & &
\end{tikzcd}$$

First-Order Challenge: Pre-Deletion Check

$$
\begin{tikzcd}
\eta_5\colon q(Y) \vdash p(a) & & {\color{red}\eta_4\colon p(X) \vdash} & & \eta_3\colon \vdash p(b), q(Y) \\
& \eta_1\colon q(Y) \vdash \arrow[ul] \arrow[ur] & & \eta_2\colon \vdash q(Y) \arrow[ul] \arrow[ur] & \\
& & \bot \arrow[ul] \arrow[ur] & &
\end{tikzcd}
$$

First-Order Challenge: Pre-Deletion Check

$$
\begin{tikzcd}[column sep=small]
\eta_5\colon q(Y) \vdash p(a) & & {\color{red}\eta_4\colon p(X) \vdash} & & \eta_3\colon \vdash p(b), q(Y) \\
& \eta_1\colon q(Y) \vdash \arrow[ul] \arrow[ur] & & \eta_2\colon \vdash q(Y) \arrow[ul] \arrow[ur] & \\
& & \bot \arrow[ul] \arrow[ur] & &
\end{tikzcd}
$$

$$
\begin{tikzcd}[column sep=small]
\eta_5'\colon q(Y) \vdash p(a) & & \eta_3'\colon \vdash p(b), q(Y) \\
& \eta\colon \vdash p(a), p(b) \arrow[ul] \arrow[ur] &
\end{tikzcd}
$$

First-Order Challenge: Pre-Deletion Check

$$
\begin{tikzcd}
\eta_5\colon q(Y) \vdash p(a) & & \color{red}{\eta_4\colon p(X) \vdash} & & \eta_3\colon \vdash p(b), q(Y) \\
& \eta_1\colon q(Y) \vdash \arrow[ul] \arrow[ur] & & \eta_2\colon \vdash q(Y) \arrow[ul] \arrow[ur] & \\
& & \bot \arrow[ul] \arrow[ur] & & \\
& \eta_5'\colon q(Y) \vdash p(a) & & \eta_3'\colon \vdash p(b), q(Y) & \\
& & \eta\colon \vdash p(a), p(b) \arrow[ul] \arrow[ur] & & \\
& & \cancel{\lfloor \eta \rfloor} \arrow[u] & &
\end{tikzcd}
$$

First-Order Challenge: Pre-Deletion Check

$$
\begin{tikzcd}
\eta_5\colon q(Y) \vdash p(a) & & \color{red}{\eta_4\colon p(X) \vdash} & & \eta_3\colon \vdash p(b), q(Y) \\
& \eta_1\colon q(Y) \vdash \arrow[ul] \arrow[ur] & & \eta_2\colon \vdash q(Y) \arrow[ul] \arrow[ur] & \\
& & \bot \arrow[ul] \arrow[ur] & &
\end{tikzcd}
$$

Definition (Pre-Deletion Property)

η unit, $l \in \eta$, such that l is resolved with literals $l_1, \ldots, l_n$ in a proof ψ. η satisfies the *pre-deletion unifiability* property in ψ if $l_1, \ldots, l_n$ and $\bar{l}$ are unifiable.

First-Order Challenge: Post-Deletion Check

$$
\begin{tikzcd}
 & \eta_1\colon r(Y), p(X, q(Y, b)), p(X, Y) \vdash & & \eta_2\colon \vdash p(U, V) \\
\eta_4\colon \vdash r(W) & & \eta_3\colon r(V), p(U, q(V, b)) \vdash \arrow[ul] \arrow[ur] & \\
 & \eta_5\colon p(U, q(W, b)) \vdash \arrow[ul] \arrow[ur] & & \\
 & & \perp \arrow[ul] \arrow[uuur, bend right] &
\end{tikzcd}
$$

First-Order Challenge: Post-Deletion Check

$$
\begin{tikzcd}
 & & \eta_1\colon r(Y), p(X, q(Y, b)), p(X, Y) \vdash & & {\color{red}\eta_2\colon \vdash p(U, V)} \\
 & \eta_4\colon \vdash r(W) & & \eta_3\colon r(V), p(U, q(V, b)) \vdash \arrow[ul] \arrow[ur] & \\
 & & \eta_5\colon p(U, q(W, b)) \vdash \arrow[ul] \arrow[ur] & & \\
 & & & \bot \arrow[ul] \arrow[uuur, bend right] &
\end{tikzcd}
$$

First-Order Challenge: Post-Deletion Check

$$\eta_1\colon r(Y), p(X, q(Y, b)), p(X, Y) \vdash$$

$$\eta_2\colon \vdash p(U, V)$$

$$\eta_4\colon \vdash r(W)$$

$$\eta_3\colon r(V), p(U, q(V, b)) \vdash$$

$$\eta_5\colon p(U, q(W, b)) \vdash$$

$$\perp$$

$$\eta_4'\colon \vdash r(W)$$

$$\eta_1'\colon r(Y), p(X, q(Y, b)), p(X, Y) \vdash$$

$$\eta_5'\colon p(X, q(W, b)), p(X, W) \vdash$$

First-Order Challenge: Post-Deletion Check

$$
\begin{tikzcd}
 & \eta_1\colon r(Y), p(X, q(Y,b)), p(X,Y) \vdash & & \color{red}{\eta_2\colon \vdash p(U,V)} \\
\eta_4\colon \vdash r(W) & & \eta_3\colon r(V), p(U, q(V,b)) \vdash \arrow[ul] \arrow[ur] & \\
 & \eta_5\colon p(U, q(W,b)) \vdash \arrow[ul] \arrow[ur] & & \\
 & & \bot \arrow[ul] \arrow[uuur, bend right] &
\end{tikzcd}
$$

$$
\begin{tikzcd}
\eta'_4\colon \vdash r(W) & & \eta'_1\colon r(Y), p(X, q(Y,b)), p(X,Y) \vdash \\
 & \eta'_5\colon p(X, q(W,b)), p(X,W) \vdash \arrow[ul] \arrow[ur] & \\
 & \lfloor \cancel{\eta'_5} \rfloor \arrow[u] &
\end{tikzcd}
$$

First-Order Challenge: Post-Deletion Check

$$
\begin{tikzcd}
 & \eta_1\colon r(Y), p(X, q(Y, b)), p(X, Y) \vdash & & \color{red}{\eta_2\colon \vdash p(U, V)} \\
\eta_4\colon \vdash r(W) & & \eta_3\colon r(V), p(U, q(V, b)) \vdash \arrow[ul] \arrow[ur] & \\
 & \eta_5\colon p(U, q(W, b)) \vdash \arrow[ul] \arrow[ur] & & \\
 & & \perp \arrow[ul] \arrow[uuur, bend right] &
\end{tikzcd}
$$

Definition (Post-Deletion Property)

η unit, $l \in \eta$, such that l is resolved with literals $l_1, \ldots, l_n$ in a proof ψ. η satisfies the *post-deletion unifiability* property in ψ if $l_1^{\dagger\downarrow}, \ldots, l_n^{\dagger\downarrow}$ and $\overline{l^{\dagger}}$ are unifiable, where $l^{\dagger}$ is the literal in $\psi' = \psi \setminus \{\eta\}$ corresponding to l in ψ, and $l^{\dagger\downarrow}$ is the descendant of $l^{\dagger}$ in the roof of ψ'.

First-Order Lower Units Challenges

- Deletion changes literals
- Unit collection depends on whether contraction is possible after propagation down the proof

Deletion of units require knowledge of proof after deletion, and deletion depends on what will be lowered.

- $O(n^2)$ solution to have full knowledge
- Difficult bookkeeping required for implementation

Greedy First-Order Lower Units - A Quicker Alternative

- Ignore post-deletion satisfaction
- Focus on pre-deletion satisfaction
- Greedy contraction

Faster run-time (linear; one traversal)
Easier to implement

- Doesn't always compress (returns original proof sometimes)

Greedy First-Order Lower Units - A Quicker Alternative

- Ignore post-deletion satisfaction
- Focus on pre-deletion satisfaction
- Greedy contraction

Faster run-time (linear; one traversal)
Easier to implement

- Doesn't always compress (returns original proof sometimes)

Greedy First-Order Lower Units - A Quicker Alternative

- Ignore post-deletion satisfaction
- Focus on pre-deletion satisfaction
- Greedy contraction

Faster run-time (linear; one traversal)
Easier to implement

- Doesn't always compress (returns original proof sometimes)

First-Order Example

$$
\begin{tikzcd}
 & & \eta_2\colon p(W) \vdash q(X), r(Y), t(Z) & & \eta_1\colon \vdash p(a) \\
 & \eta_4\colon r(X), p(V) \vdash q(Y), t(Z) & & \eta_3\colon \vdash q(X), r(Y), t(Z) \arrow[ul] \arrow[ur] & \\
 & & \eta_5\colon p(V) \vdash q(X), t(Z) \arrow[ul] \arrow[ur] & & \\
 & \eta_7\colon q(X), p(a) \vdash t(Z) & & \eta_6\colon \vdash q(X), t(Z) \arrow[ul] \arrow[uuur, bend right] & \\
 \eta_9\colon t(Z) \vdash & & \eta_8\colon p(a) \vdash t(Z) \arrow[ul] \arrow[ur] & & \\
 & \eta_{10}\colon p(a) \vdash \arrow[ul] \arrow[ur] & & & \\
 & & \bot \arrow[ul] \arrow[uuuuuurr, bend right] & &
\end{tikzcd}
$$

First-Order Example

η_2: $p(W) \vdash q(X), r(Y), t(Z)$ $\quad\quad$ η_1: $\vdash p(a)$

η_4: $r(X), p(V) \vdash q(Y), t(Z)$ $\quad\quad$ η_3: $\vdash q(X), r(Y), t(Z)$

η_5: $p(V) \vdash q(X), t(Z)$

η_7: $q(X), p(a) \vdash t(Z)$ $\quad\quad$ η_6: $\vdash q(X), t(Z)$

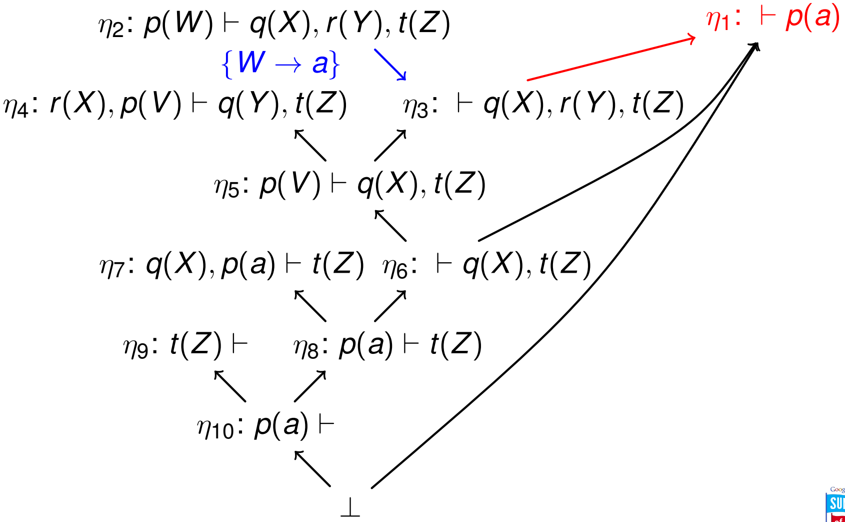
η_9: $t(Z) \vdash$ $\quad\quad$ η_8: $p(a) \vdash t(Z)$

η_{10}: $p(a) \vdash$

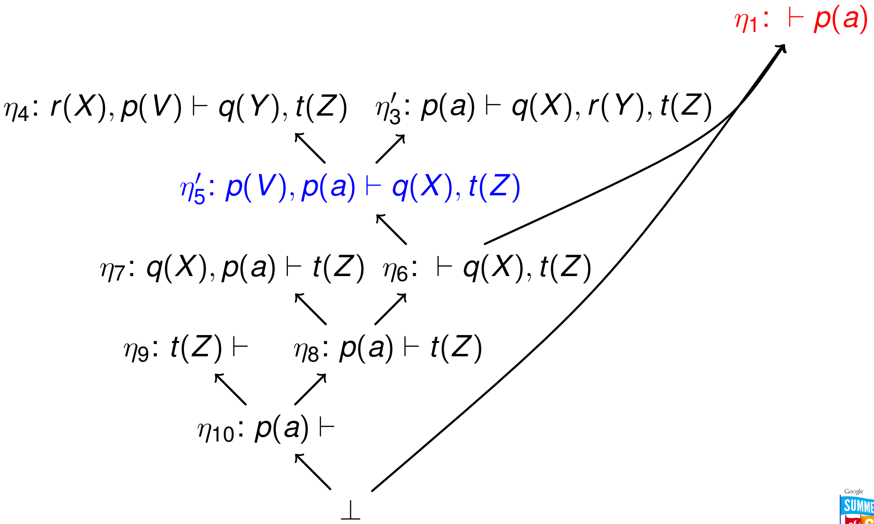
$$\perp$$

$$\sigma = \{W \to a, V \to a\}$$

First-Order Example



First-Order Example



First-Order Example

$$\eta_1 \colon \ \vdash p(a)$$

$$\eta_4 \colon r(X), p(V) \vdash q(Y), t(Z) \qquad \eta_3 \colon p(a) \vdash q(X), r(Y), t(Z)$$

$$\eta_5' \colon p(V), p(a) \vdash q(X), t(Z)$$

$$\{V \rightarrow a\}$$

$$\eta_7 \colon q(X), p(a) \vdash t(Z) \qquad \eta_6 \colon \ \vdash q(X), t(Z)$$

$$\eta_9 \colon t(Z) \vdash \qquad \eta_8 \colon p(a) \vdash t(Z)$$

$$\eta_{10} \colon p(a) \vdash$$

$$\perp$$

First-Order Example

$$
\begin{tikzcd}[column sep=small]
 & & & & & \eta_1\colon\ \vdash p(a) \\
 & \eta_4\colon r(X), p(V) \vdash q(Y), t(Z) & & & \eta_3\colon p(a) \vdash q(X), r(Y), t(Z) & \\
 & & & & \eta_6'\colon p(a), p(a) \vdash q(X), t(Z) \arrow[ulll] \arrow[u] & \\
 & & \eta_7\colon q(X), p(a) \vdash t(Z) & & \lfloor \eta_6' \rfloor\colon p(a) \vdash q(X), t(Z) \arrow[u] & \\
 & \eta_9\colon t(Z) \vdash & & \eta_8\colon p(a) \vdash t(Z) \arrow[ul] \arrow[ur] & & \\
 & & \eta_{10}\colon p(a) \vdash \arrow[ul] \arrow[ur] & & & \\
 & & & \bot \arrow[ul] \arrow[uuuuuurr, bend right] & &
\end{tikzcd}
$$

First-Order Example

$\eta_1 \colon \vdash p(a)$

$\eta_4 \colon r(X), p(V) \vdash q(Y), t(Z)$ $\quad$ $\eta_3 \colon p(a) \vdash q(X), r(Y), t(Z)$

$\eta_6' \colon p(a), p(a) \vdash q(X), t(Z)$

$\eta_7 \colon q(X), p(a) \vdash t(Z)$ $\quad$ $\lfloor \eta_6' \rfloor \colon p(a) \vdash q(X), t(Z)$

$\eta_9 \colon t(Z) \vdash$ $\quad$ $\eta_8 \colon p(a) \vdash t(Z)$

$\eta_{10} \colon p(a) \vdash$

$\emptyset$

$\perp$

First-Order Example

$\eta_1\colon \vdash p(a)$

$$
\begin{tikzcd}
\eta_4\colon r(X), p(V) \vdash q(Y), t(Z) & & \eta_3\colon p(a) \vdash q(X), r(Y), t(Z) \\
& & \eta_6'\colon p(a), p(a) \vdash q(X), t(Z) \arrow[ull] \arrow[u] \\
\eta_7\colon q(X), p(a) \vdash t(Z) & & \lfloor \eta_6' \rfloor\colon p(a) \vdash q(X), t(Z) \arrow[u] \\
\eta_9\colon t(Z) \vdash & \eta_8\colon p(a) \vdash t(Z) \arrow[ul] \arrow[ur] & \\
& \eta_{10}\colon p(a) \vdash \arrow[ul] \arrow[u] &
\end{tikzcd}
$$

First-Order Example

$$\eta_1\colon\ \vdash p(a)$$

$$\eta_4\colon r(X), p(V) \vdash q(Y), t(Z) \qquad \eta_3\colon p(a) \vdash q(X), r(Y), t(Z)$$

$$\eta_6'\colon p(a), p(a) \vdash q(X), t(Z)$$

$$\eta_7\colon q(X), p(a) \vdash t(Z) \qquad \lfloor \eta_6' \rfloor\colon p(a) \vdash q(X), t(Z)$$

$$\eta_9\colon t(Z) \vdash \qquad \eta_8\colon p(a) \vdash t(Z)$$

$$\eta_{10}\colon p(a) \vdash$$

$$\bot$$

Recycling Pivots

Removes *irregularities*: inferences η where the pivot occurs as a pivot of another inference below η on the path to the root

- Store a set of *safe* $\mathcal{S}(\eta)$ literals for each node η
- If there are multiple paths, take *intersection* of safe literals
- Bottom-up: compute safe literals; mark deletions
- Top-down: regularize

Recycling Pivots

Removes *irregularities*: inferences η where the pivot occurs as a pivot of another inference below η on the path to the root

- Store a set of *safe* $\mathcal{S}(\eta)$ literals for each node η
- If there are multiple paths, take *intersection* of safe literals
- Bottom-up: compute safe literals; mark deletions
- Top-down: regularize

Recycling Pivots

Removes *irregularities*: inferences η where the pivot occurs as a pivot of another inference below η on the path to the root

- Store a set of *safe* $\mathcal{S}(\eta)$ literals for each node η
- If there are multiple paths, take *intersection* of safe literals
- Bottom-up: compute safe literals; mark deletions
- Top-down: regularize

Regularization Can Be Bad

Resolution without irregularities is still complete. But:

Theorem ([Tse70])

There are unsatisfiable formulas whose shortest regular resolution refutations are exponentially longer than their shortest unrestricted resolution refutations.

Propositional Example

$$
\begin{tikzcd}[column sep=small]
\bar{b} & & \bar{a}b & \\
& \bar{a} \arrow[ul] \arrow[ur] & & a\bar{b} \\
b & & \bar{b} \arrow[ul] \arrow[ur] & \\
& \bot \arrow[ul] \arrow[ur] & &
\end{tikzcd}
$$

$$
\begin{tikzcd}[column sep=small]
\bar{b} & & \bar{a}b & \\
& \bar{a} \arrow[ul] \arrow[ur] & & a\bar{b} \\
b & & \bar{b} \arrow[ul] \arrow[ur] & \\
& \bot \arrow[ul] \arrow[ur] & &
\end{tikzcd}
$$

$\uparrow$ safe literal collection

Propositional Example

$$
\begin{tikzcd}
\bar{b} & & \bar{a}b & \\
& \bar{a} \arrow[ul] \arrow[ur] & & a\bar{b} \\
b & & \bar{b} \arrow[ul] \arrow[ur] & \\
& \bot \arrow[ul] \arrow[ur] & &
\end{tikzcd}
\qquad
\begin{tikzcd}
& \bar{b} & & \bar{a}b & & \\
& & \bar{a} \arrow[ul] \arrow[ur] & & a\bar{b} & \\
\{b\} & b & & \bar{b} \arrow[ul] \arrow[ur] & & \{\bar{b}\} \\
& & \bot \arrow[ul] \arrow[ur] & & &
\end{tikzcd}
$$

$\uparrow$ safe literal collection

Propositional Example

$$
\begin{tikzcd}
\bar{b} & & \bar{a}b & \\
 & \bar{a} \arrow[ul] \arrow[ur] & & a\bar{b} \\
b & & \bar{b} \arrow[ul] \arrow[ur] & \\
 & \perp \arrow[ul] \arrow[ur] & &
\end{tikzcd}
$$

$$
\begin{tikzcd}
 & \bar{b} & & \bar{a}b & & \\
 & \{\bar{a}\bar{b}\} & \bar{a} \arrow[ul] \arrow[ur] & & a\bar{b} & \{a\bar{b}\} \\
\{b\} & b & & \bar{b} \arrow[ul] \arrow[ur] & \{\bar{b}\} & \\
 & & \perp \arrow[ul] \arrow[ur] & & &
\end{tikzcd}
$$

$\uparrow$ safe literal collection

Propositional Example

$$
\begin{tikzcd}
\bar{b} & & \bar{a}b & \\
& \bar{a} \arrow[ul] \arrow[ur] & & a\bar{b} \\
b & & \bar{b} \arrow[ul] \arrow[ur] & \\
& \bot \arrow[ul] \arrow[ur] & &
\end{tikzcd}
$$

$$
\begin{tikzcd}
\{\bar{a}{\color{red}\bar{b}}\} & \bar{b} & & \bar{a}b & \{ab\} & \\
& \{\bar{a}\bar{b}\} & \bar{a} \arrow[ul] \arrow[ur] & & a\bar{b} & \{a\bar{b}\} \\
\{b\} & b & & \bar{b} \arrow[ul] \arrow[ur] & \{\bar{b}\} & \\
& & \bot \arrow[ul] \arrow[ur] & & &
\end{tikzcd}
$$

$\uparrow$ safe literal collection

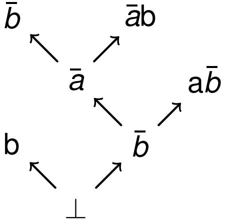
Propositional Example

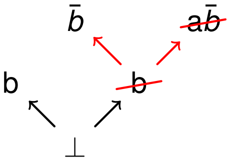
$$
\begin{tikzcd}
\bar{b} & & \bar{a}b & \\
& \bar{a} \arrow[ul] \arrow[ur] & & a\bar{b} \\
b & & \bar{b} \arrow[ul] \arrow[ur] & \\
& \bot \arrow[ul] \arrow[ur] & &
\end{tikzcd}
$$

$$
\begin{tikzcd}
\{\bar{a}\bar{b}\} & & \bar{b} & & \cancel{\bar{a}b} & & \{ab\} \\
& \{\bar{a}\bar{b}\} & & \bar{a} \arrow[ul] \arrow[ur] & & a\bar{b} & \{a\bar{b}\} \\
\{b\} & & b & & \bar{b} \arrow[ul] \arrow[ur] & & \{\bar{b}\} \\
& & & \bot \arrow[ul] \arrow[ur] & & &
\end{tikzcd}
$$

↑ safe literal collection

Propositional Example





$\downarrow$ regularization

Propositional Example

$$
\begin{tikzcd}
\bar{b} & & \bar{a}b & \\
& \bar{a} \arrow[ul] \arrow[ur] & & a\bar{b} \\
b & & \bar{b} \arrow[ul] \arrow[ur] & \\
& \bot \arrow[ul] \arrow[ur] & &
\end{tikzcd}
$$

$$
\begin{tikzcd}
b & & \bar{b} \\
& \bot \arrow[ul] \arrow[ur] &
\end{tikzcd}
$$

$\downarrow$ regularization

Pre-Regularization Checks I

$$
\begin{tikzcd}
\begin{matrix} \eta_1\colon \vdash p(W,X) \\ \{\vdash q(c), \textcolor{red}{p(a,X)}\,\} \end{matrix} & & \begin{matrix} \eta_2\colon p(W,X) \vdash q(c) \\ \{p(W,X) \vdash q(c), p(a,X)\} \end{matrix} \\
\begin{matrix} \eta_3\colon \vdash q(c) \\ \{\vdash q(c), p(a,X)\} \end{matrix} \arrow[u] \arrow[urr] & & \begin{matrix} \eta_4\colon q(c) \vdash p(a,X) \\ \{q(c) \vdash p(a,X)\} \end{matrix} \\
\begin{matrix} \eta_6\colon p(Y,b) \vdash \\ \{p(Y,b) \vdash\} \end{matrix} & & \begin{matrix} \eta_5\colon \vdash p(a,X) \\ \{\vdash p(a,X)\} \end{matrix} \arrow[ull] \arrow[u] \\
& \bot \arrow[ul] \arrow[ur] &
\end{tikzcd}
$$

$$\sigma = \{W \to a\} \implies \sigma\eta_1 \in \mathcal{S}(\eta_1)$$

Pre-Regularization Checks I

$$\begin{tikzcd}
\eta_6\colon p(Y,b) \vdash & & \eta_1\colon \vdash p(W,X) \\
& \perp \arrow[ul] \arrow[ur] &
\end{tikzcd}$$

$$\sigma = \{W \to Y, X \to b\}$$

Pre-Regularization Checks II

$$
\begin{tikzcd}
\eta_1\colon \vdash p(W,c) & & \eta_2\colon p(W,X) \vdash q(c) \\
& \eta_3\colon \vdash q(c) \arrow[ul] \arrow[ur] & & \eta_4\colon q(c) \vdash p(a,X) \\
\eta_6\colon p(Y,b) \vdash & & \eta_5\colon \vdash p(a,X) \arrow[ul] \arrow[ur] \\
& \bot \arrow[ul] \arrow[ur]
\end{tikzcd}
$$

$\{\vdash q(c), p(a,X)\}$ (at η_1)

$\{p(W,X) \vdash q(c), p(a,X)\}$ (at η_2)

$\{\vdash q(c), p(a,X)\}$ (at η_3)

$\{q(c) \vdash p(a,X)\}$ (at η_4)

$\{p(Y,b) \vdash\}$ (at η_6)

$\{\vdash p(a,X)\}$ (at η_5)

$$\sigma = \{W \to a, X \to c\} \implies \sigma\eta_1 \in \mathcal{S}(\eta_1)$$

but...

Pre-Regularization Checks II

$$\begin{tikzcd} \eta_6\colon p(Y,b) \vdash & & \eta_1\colon \vdash p(c,a) \\ & \cancel{\bot} \arrow[ul] \arrow[ur] & \end{tikzcd}$$

no σ!

Pre-Regularization Unifiability

Definition

Let η be a node with pivot ℓ' unifiable with safe literal ℓ which is resolved against literals $\ell_1, \ldots, \ell_n$ in a proof ψ. η is said to satisfy the *pre-regularization unifiability property* in ψ if $\ell_1, \ldots, \ell_n$, and $\bar{\ell}'$ are unifiable.

Post-Regularization Checks

$$\begin{tikzcd}
\eta_1\colon p(U,V) \vdash q(f(a,V),U) & & \eta_2\colon q(f(a,X),Y), q(T,X) \vdash q(f(a,Z),Y) \\
\eta_4\colon \vdash q(R,S) & & \eta_3\colon p(U,V), q(T,V) \vdash q(f(a,Z),U) \arrow[ull] \arrow[u] \\
\eta_6\colon \vdash p(c,d) & & \eta_5\colon p(U,V) \vdash q(f(a,Z),U) \arrow[ull] \arrow[u] \\
\eta_8\colon q(f(a,e),c) \vdash & & \eta_7\colon \vdash q(f(a,Z),c) \arrow[ull] \arrow[u] \\
& \bot \arrow[ul] \arrow[ur] &
\end{tikzcd}$$

$$\mathcal{S}(\eta_3) = \{q(T,V), p(c,d) \vdash q(f(a,e),c)\}$$

Post-Regularization Checks

$$
\begin{tikzcd}
\eta_1 \colon p(U,V) \vdash q(f(a,V),U) & & \cancel{\eta_2 \colon q(f(a,X),Y), q(T,X) \vdash q(f(a,Z),Y)} \\
\eta_4 \colon \vdash q(R,S) & & \eta_3 \colon p(U,V), q(T,V) \vdash q(f(a,Z),U) \arrow[ull] \arrow[u] \\
\eta_6 \colon \vdash p(c,d) & & \eta_5 \colon p(U,V) \vdash q(f(a,Z),U) \arrow[ull] \arrow[u] \\
\eta_8 \colon q(f(a,e),c) \vdash & & \eta_7 \colon \vdash q(f(a,Z),c) \arrow[ull] \arrow[u] \\
& \bot \arrow[ul] \arrow[ur] &
\end{tikzcd}
$$

Post-Regularization Checks

$$
\begin{tikzcd}
\eta_4 \colon \vdash q(R, S) & & \eta_1 \colon p(U, V) \vdash q(f(a, V), U) \\
\eta_6 \colon \vdash p(c, d) & & \eta_5 \colon p(U, V) \vdash q(f(a, Z), U) \arrow[ull] \arrow[u] \\
\eta_8 \colon q(f(a, e), c) \vdash & & \eta_7 \colon \vdash q(f(a, Z), c) \arrow[ull] \arrow[u] \\
& \bot \arrow[ul] \arrow[ur] &
\end{tikzcd}
$$

Post-Regularization Checks

$$
\begin{tikzcd}
\cancel{\eta_4 \colon \vdash q(R, S)} & & \eta_1 \colon p(U, V) \vdash q(f(a, V), U) \\
\eta_6 \colon \vdash p(c, d) & & \eta_5 \colon p(U, V) \vdash q(f(a, Z), U) \arrow[ull] \arrow[u] \\
\eta_8 \colon q(f(a, e), c) \vdash & & \eta_7 \colon \vdash q(f(a, Z), c) \arrow[ull] \arrow[u] \\
& \bot \arrow[ul] \arrow[ur] &
\end{tikzcd}
$$

Post-Regularization Checks

$$
\begin{tikzcd}
 & \eta_6\colon \vdash p(c,d) & & {\color{blue}\eta_1\colon p(U,V) \vdash q(f(a,V),U)} \\
\eta_8\colon q(f(a,e),c) \vdash & & \eta_7\colon \vdash q(f(a,Z),c) \arrow[ul] \arrow[ur] & \\
 & \bot \arrow[ul] \arrow[ur] & &
\end{tikzcd}
$$

Post-Regularization Checks

$$
\begin{tikzcd}
\eta_6\colon \vdash p(c,d) & & \eta_1\colon p(U,V) \vdash q(f(a,V),U) \\
\eta_8\colon q(f(a,e),c) \vdash & & \eta_7\colon \vdash q(f(a,{\color{blue}d}),c) \arrow[ull] \arrow[u] \\
& \cancel{\perp} \arrow[ul] \arrow[ur] &
\end{tikzcd}
$$

Regularization Unifiability

Definition

Let η be a node with safe literals $\mathcal{S}(\eta) = \phi$ that is marked for regularization with parents η_1 and η_2, where η_2 is marked as a `deletedNode` in a proof ψ. η is said to satisfy the *regularization unifiability property* in ψ if there exists a substitution σ such that $\eta_1\sigma \subseteq \phi$.

Experiment Setup

- Simple First-Order Lower units implemented as part of Skeptik (in Scala)
- 308 real first-order proofs generated by SPASS from problems from TPTP Problem Library
  - 2280 initial problems (1032 known unsatisfiable)
  - SPASS asked to use only resolution and contraction rules
  - 300s timeout
- proofs *generated* on cluster at the University of Victoria
- proofs *compressed* on *this laptop*

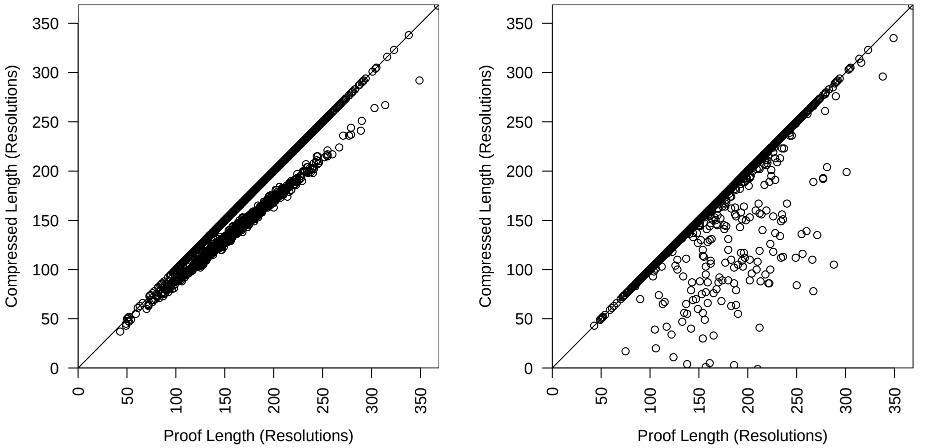
Time to generate proofs: $\approx$ 40 minutes
Time to compress proofs: $\approx$ 5 seconds

Experiment Setup

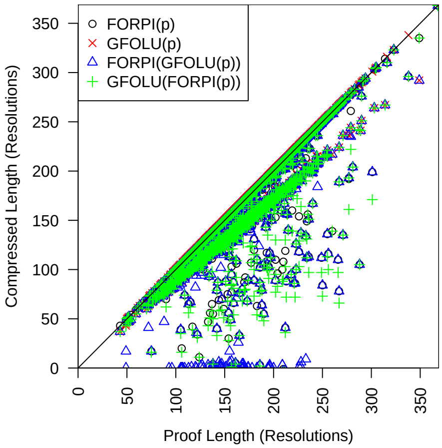
- Simple First-Order Lower units implemented as part of Skeptik (in Scala)
- 308 real first-order proofs generated by SPASS from problems from TPTP Problem Library
  - 2280 initial problems (1032 known unsatisfiable)
  - SPASS asked to use only resolution and contraction rules
  - 300s timeout
- proofs *generated* on cluster at the University of Victoria
- proofs *compressed* on *this laptop*

Time to generate proofs: $\approx$ 40 minutes
Time to compress proofs: $\approx$ 5 seconds

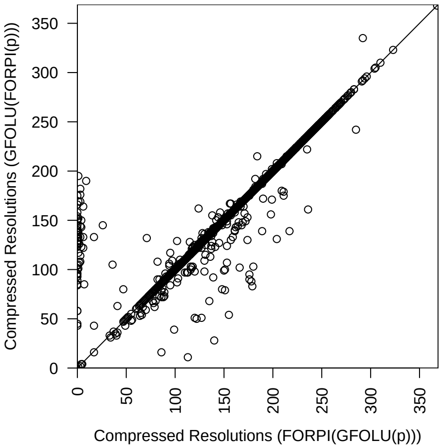
Results


Compressed Length (Resolutions)
Proof Length (Resolutions)
0
50
100
150
200
250
300
350
Compressed Length (Resolutions)
Proof Length (Resolutions)

Results



FORPI(p)
GFOLU(p)
FORPI(GFOLU(p))
GFOLU(FORPI(p))
Compressed Length (Resolutions)
Proof Length (Resolutions)
0
50
100
150
200
250
300
350



Compressed Resolutions (GFOLU(FORPI(p)))
Compressed Resolutions (FORPI(GFOLU(p)))
0
50
100
150
200
250
300
350

Results I

Percent of proofs compressed:

- LU(p): 36%
- RPI(p): 9%
- RPI(LU(p)): 43%
- LU(RPI(p)): 42%

Results II

Successful cumulative compression ratio:

- LU(p): 0.95
- RPI(p): 0.72
- RPI(LU(p)): 0.85
- LU(RPI(p)): 0.89

Conclusion

- Two simple, quick algorithms lifted from propositional to first-order logic for proof compression
  - LowerUnits compresses more often
  - RPI compresses more
- Future work:
  - Explore other proof compression algorithms?
  - Explore ways of dealing with the post-deletion property quickly

Thank you for your attention.
Any questions?

- Source code: `https://github.com/jgorzny/Skeptik`
- Data: `https://cs.uwaterloo.ca/~jgorzny/data/`

References I

- Omer Bar-Ilan, Oded Fuhrmann, Shlomo Hoory, Ohad Shacham, and Ofer Strichman, *Linear-time reductions of resolution proofs*, Haifa Verification Conference, Springer, 2008, pp. 114–128.
- Pascal Fontaine, Stephan Merz, and Bruno Woltzenlogel Paleo, *Compression of propositional resolution proofs via partial regularization*, International Conference on Automated Deduction, Springer, 2011, pp. 237–251.
- Marijn J. H. Heule, Oliver Kullmann, and Victor W. Marek, *Solving and verifying the boolean pythagorean triples problem via cube-and-conquer*, CoRR **abs/1605.00723** (2016).
- Rüdger Thiele, *Hilbert's twenty-fourth problem*, The American mathematical monthly **110** (2003), no. 1, 1–24.
- Gregory Tseitin, *On the complexity of proofs in propositional logics*, Seminars in Mathematics, vol. 8, 1970, pp. 466–483.

To-do

$$\cfrac{\cfrac{\cfrac{\cfrac{\eta_8\colon p(X), q(X), r(X) \vdash \qquad \eta_7\colon \vdash p(Y)}{\eta_6\colon q(Y), r(Y) \vdash} \qquad \eta_5\colon p(Z) \vdash q(Z)}{\eta_4\colon p(Z), r(Z) \vdash} \qquad \eta_3\colon \vdash r(W)}{\eta_2\colon p(W) \vdash} \qquad \eta_1\colon \vdash p(U)}{\psi\colon \bot}$$