Proof Skolemization and De-Skolemization

Matthias Baaz, Stefan Hetzl, Daniel Weller

Workshop on Logic and Computation, Vienna, 30 June 2009

Preliminaries

- classical logic, restrict to $\vee, \exists, \neg$
- tree-like **LK**-proofs

$$\frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, A \vee B}\ \vee_r^1 \qquad \frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, B \vee A}\ \vee_r^2$$

$$\frac{A, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \neg A}\ \neg_r \qquad \frac{\Gamma \vdash \Delta, A}{\neg A, \Gamma \vdash \Delta}\ \neg_l$$

$$\frac{\Gamma \vdash \Delta, A[x \leftarrow t]}{\Gamma \vdash \Delta, (\exists x)A}\ \exists_r \qquad \frac{A[x \leftarrow \alpha], \Gamma \vdash \Delta}{(\exists x)A, \Gamma \vdash \Delta}\ \exists_l$$

$$\frac{A, \Gamma \vdash \Delta \quad B, \Pi \vdash \Lambda}{A \vee B, \Gamma, \Pi \vdash \Delta, \Lambda}\ \vee_l \qquad \frac{\Gamma \vdash \Delta, A \quad A, \Pi \vdash \Lambda}{\Gamma, \Pi \vdash \Delta, \Lambda}\ cut$$

$$\frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, A}\ w_r \qquad \frac{\Gamma \vdash \Delta}{A, \Gamma \vdash \Delta}\ w_l \qquad \frac{A, A, \Gamma \vdash \Delta}{A, \Gamma \vdash \Delta}\ c_l \qquad \frac{\Gamma \vdash \Delta, A, A}{\Gamma \vdash \Delta, A}\ c_r$$

Preliminaries

- classical logic, restrict to $\vee, \exists, \neg$
- tree-like **LK**-proofs
- ancestors, descendents

$$\frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, A \vee B}\ \vee_r^1$$

$$\frac{A, \Gamma \vdash \Delta \quad B, \Pi \vdash \Lambda}{A \vee B, \Gamma, \Pi \vdash \Delta, \Lambda}\ \vee_l \qquad \frac{\Gamma \vdash \Delta, A \quad A, \Pi \vdash \Lambda}{\Gamma, \Pi \vdash \Delta, \Lambda}\ cut$$

$$\frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, A}\ w_r \qquad \frac{\Gamma \vdash \Delta, A, A}{\Gamma \vdash \Delta, A}\ c_r$$

Preliminaries

- classical logic, restrict to $\vee, \exists, \neg$
- tree-like **LK**-proofs
- ancestors, descendents
- Proof length $l(\pi) =$ number of sequents in π

Preliminaries

- classical logic, restrict to $\vee, \exists, \neg$
- tree-like **LK**-proofs
- ancestors, descendents
- Proof length $l(\pi)$ = number of sequents in π
- Polarities, strong quantifiers, strong quantifier rules

Preliminaries

- classical logic, restrict to $\vee, \exists, \neg$
- tree-like **LK**-proofs
- ancestors, descendents
- Proof length $l(\pi) =$ number of sequents in π
- Polarities, strong quantifiers, strong quantifier rules
  - If $(\exists x)F$, $G \vee H$ have positive (negative) polarity, then F, G, H have positive (negative) polarity

Preliminaries

- classical logic, restrict to $\vee, \exists, \neg$
- tree-like **LK**-proofs
- ancestors, descendents
- Proof length $l(\pi) =$ number of sequents in π
- Polarities, strong quantifiers, strong quantifier rules
  - If $(\exists x)F$, $G \vee H$ have positive (negative) polarity, then F, G, H have positive (negative) polarity
  - If $\neg F$ has positive (negative) polarity, then F has negative (positive) polarity

Preliminaries

- classical logic, restrict to $\vee, \exists, \neg$
- tree-like **LK**-proofs
- ancestors, descendents
- Proof length $l(\pi)$ = number of sequents in π
- Polarities, strong quantifiers, strong quantifier rules
  - If $(\exists x)F$, $G \vee H$ have positive (negative) polarity, then F, G, H have positive (negative) polarity
  - If $\neg F$ has positive (negative) polarity, then F has negative (positive) polarity
  - In $\Gamma \vdash \Delta$, Γ has negative and Δ has positive polarity

Preliminaries

- classical logic, restrict to $\vee, \exists, \neg$
- tree-like **LK**-proofs
- ancestors, descendents
- Proof length $l(\pi)$ = number of sequents in π
- Polarities, strong quantifiers, strong quantifier rules
  - If $(\exists x)F$, $G \vee H$ have positive (negative) polarity, then F, G, H have positive (negative) polarity
  - If $\neg F$ has positive (negative) polarity, then F has negative (positive) polarity
  - In $\Gamma \vdash \Delta$, Γ has negative and Δ has positive polarity
  - Negative $(\exists x)$ are called *strong*, positive $(\exists x)$ are called *weak*

Preliminaries

- classical logic, restrict to $\vee, \exists, \neg$
- tree-like **LK**-proofs
- ancestors, descendents
- Proof length $l(\pi)$ = number of sequents in π
- Polarities, strong quantifiers, strong quantifier rules
  - If $(\exists x)F$, $G \vee H$ have positive (negative) polarity, then F, G, H have positive (negative) polarity
  - If $\neg F$ has positive (negative) polarity, then F has negative (positive) polarity
  - In $\Gamma \vdash \Delta$, Γ has negative and Δ has positive polarity
  - Negative $(\exists x)$ are called *strong*, positive $(\exists x)$ are called *weak*
  - Rules of **LK** preserve polarity w.r.t. ancestors

Why Skolemize?

- Eliminate one type of quantifier
- Often: functions obtained have natural interpretation

Example

$$\begin{array}{ll} & (\forall x)(\exists y)(PRIME(y) \wedge DIV(y,x)) \\ \rightsquigarrow_{\mathrm{sk}} & (\forall x)(PRIME(f(x)) \wedge DIV(f(x),x)) \end{array}$$

$f(x)$ is a prime divisor of x

Different Methods

- F a closed formula
- Prefix Skolemization $\mathrm{psk}(F)$ (if F occurs positively)
  - compute prefix form F_P of F
  - if $F_i = (\exists x_1 \ldots x_n)(\forall y)F(y)$ then $F_{i+1} = (\exists x_1 \ldots x_n)F(f(x_1, \ldots, x_n))$
- Structural Skolemization $\mathrm{ssk}(F)$
  - Skolemize "in place"
  - if $(\exists y)$ is a strong quantifier in F_i in the scope of weak quantifiers $(\exists x_1), \ldots, (\exists x_n)$, then F_{i+1} is obtained from F_i by dropping $(\exists y)$ and substituting $f(x_1, \ldots, x_n)$ for y
  - Unique up to renaming of Skolem functions
- Andrews Skolemization $\mathrm{ask}(F)$
  - substitute $f(x_{i_1}, \ldots, x_{i_k})$ for y, where the x_{i_j} are in the scope of $(\exists y)$

Different Methods

- F a closed formula
- Prefix Skolemization $\mathrm{psk}(F)$ (if F occurs positively)
  - compute prefix form F_P of F
  - if $F_i = (\exists x_1 \ldots x_n)(\forall y)F(y)$ then $F_{i+1} = (\exists x_1 \ldots x_n)F(f(x_1, \ldots, x_n))$
- Structural Skolemization $\mathrm{ssk}(F)$
  - Skolemize "in place"
  - if $(\exists y)$ is a strong quantifier in F_i in the scope of weak quantifiers $(\exists x_1), \ldots, (\exists x_n)$, then F_{i+1} is obtained from F_i by dropping $(\exists y)$ and substituting $f(x_1, \ldots, x_n)$ for y
  - Unique up to renaming of Skolem functions
- Andrews Skolemization $\mathrm{ask}(F)$
  - substitute $f(x_{i_1}, \ldots, x_{i_k})$ for y, where the x_{i_j} are in the scope of $(\exists y)$

Different Methods

- F a closed formula
- Prefix Skolemization $\mathrm{psk}(F)$ (if F occurs positively)
  - compute prefix form F_P of F
  - if $F_i = (\exists x_1 \dots x_n)(\forall y)F(y)$ then $F_{i+1} = (\exists x_1 \dots x_n)F(f(x_1, \dots, x_n))$
- Structural Skolemization $\mathrm{ssk}(F)$
  - Skolemize "in place"
  - if $(\exists y)$ is a strong quantifier in F_i in the scope of weak quantifiers $(\exists x_1), \dots, (\exists x_n)$, then F_{i+1} is obtained from F_i by dropping $(\exists y)$ and substituting $f(x_1, \dots, x_n)$ for y
  - Unique up to renaming of Skolem functions
- Andrews Skolemization $\mathrm{ask}(F)$
  - substitute $f(x_{i_1}, \dots, x_{i_k})$ for y, where the x_{i_j} are in the scope of $(\exists y)$

Different Methods

- F a closed formula
- Prefix Skolemization $\mathrm{psk}(F)$ (if F occurs positively)
  - compute prefix form F_P of F
  - if $F_i = (\exists x_1 \dots x_n)(\forall y)F(y)$ then $F_{i+1} = (\exists x_1 \dots x_n)F(f(x_1, \dots, x_n))$
- Structural Skolemization $\mathrm{ssk}(F)$
  - Skolemize "in place"
  - if $(\exists y)$ is a strong quantifier in F_i in the scope of weak quantifiers $(\exists x_1), \dots, (\exists x_n)$, then F_{i+1} is obtained from F_i by dropping $(\exists y)$ and substituting $f(x_1, \dots, x_n)$ for y
  - Unique up to renaming of Skolem functions
- Andrews Skolemization $\mathrm{ask}(F)$
  - substitute $f(x_{i_1}, \dots, x_{i_k})$ for y, where the x_{i_j} are in the scope of $(\exists y)$

Different Methods

- F a closed formula
- Prefix Skolemization $\mathrm{psk}(F)$ (if F occurs positively)
  - compute prefix form F_P of F
  - if $F_i = (\exists x_1 \dots x_n)(\forall y)F(y)$ then $F_{i+1} = (\exists x_1 \dots x_n)F(f(x_1, \dots, x_n))$
- Structural Skolemization $\mathrm{ssk}(F)$
  - Skolemize "in place"
  - if $(\exists y)$ is a strong quantifier in F_i in the scope of weak quantifiers $(\exists x_1), \dots, (\exists x_n)$, then F_{i+1} is obtained from F_i by dropping $(\exists y)$ and substituting $f(x_1, \dots, x_n)$ for y
  - Unique up to renaming of Skolem functions
- Andrews Skolemization $\mathrm{ask}(F)$
  - substitute $f(x_{i_1}, \dots, x_{i_k})$ for y, where the x_{i_j} are in the scope of $(\exists y)$

Different Methods

- F a closed formula
- Prefix Skolemization $\mathrm{psk}(F)$ (if F occurs positively)
  - compute prefix form F_P of F
  - if $F_i = (\exists x_1 \ldots x_n)(\forall y)F(y)$ then $F_{i+1} = (\exists x_1 \ldots x_n)F(f(x_1, \ldots, x_n))$
- Structural Skolemization $\mathrm{ssk}(F)$
  - Skolemize “in place”
  - if $(\exists y)$ is a strong quantifier in F_i in the scope of weak quantifiers $(\exists x_1), \ldots, (\exists x_n)$, then F_{i+1} is obtained from F_i by dropping $(\exists y)$ and substituting $f(x_1, \ldots, x_n)$ for y
  - Unique up to renaming of Skolem functions
- Andrews Skolemization $\mathrm{ask}(F)$
  - substitute $f(x_{i_1}, \ldots, x_{i_k})$ for y, where the x_{i_j} are in the scope of $(\exists y)$

Different Methods

Definition (Herbrand complexity)

If S is a valid sequent containing only weak quantifiers, then $\mathrm{HC}(S)$ is the minimal length of a Herbrand sequent of S.

Theorem (M. Baaz, A. Leitsch 1994)

There exists a sequence of sequents (S_n) *such that for some prefix Skolemization* $\mathrm{psk}(S_n)$, $\mathrm{HC}(\mathrm{psk}(S_n))$ *is non-elementary but* $\mathrm{HC}(\mathrm{ssk}(S_n))$ *is elementary.*

Proposition (M. Baaz, A. Leitsch 1994)

Let S' *be obtained from a sequent* S *by antiprenexing via quantifier shifting. Then* $\mathrm{HC}(\mathrm{ssk}(S')) \leq \mathrm{HC}(\mathrm{ssk}(S))$.

Proof Skolemization

Problem (Proof Skolemization)

Input: **LK**-*proof of* S
Output: **LK**-*proof of* $\mathrm{ssk}(S)$

Proof Skolemization

Problem (Proof Skolemization)

Input: **LK**-*proof of* S
Output: **LK**-*proof of* $\mathrm{ssk}(S)$

Proposition (M. Baaz, A. Leitsch 1999)

Let π *be an* **LK**-*proof of* S. *Then there exists an* **LK**-*proof* π_{ssk} *of* $\mathrm{ssk}(S)$ *s.t.* $l(\pi_{\mathrm{ssk}}) \leq l(\pi)$.

Proof Skolemization

Proof sketch.

Assume S contains negative occurrence of $(\exists x)A(x)$. It can be introduced in π as

1. $$\frac{A(\alpha), \Gamma \vdash \Delta}{(\exists x)A(x), \Gamma \vdash \Delta}\ \exists_l$$
2. $$\frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, A \vee B}\ \vee_r^1 \qquad \frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, B \vee A}\ \vee_r^2$$
   s.t. $(\exists x)A(x)$ is a subformula of B
3. $$\frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, B}\ w_r \quad \frac{\Gamma \vdash \Delta}{B, \Gamma \vdash \Delta}\ w_l$$
   s.t. $(\exists x)A(x)$ is a subformula of B

Proof Skolemization

Proof sketch.

$$\frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, A \vee B[(\exists x)A(x)]} \vee_r^1$$

replaced by

$$\frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, A \vee B[A(t)]} \vee_r^1$$

where $A(t)$ is Skolemization of $(\exists x)A(x)$ in S.
Modify descendents of $A \vee B$ appropriately.

Proof Skolemization

Proof sketch.

$$\frac{A(\alpha), \Gamma \vdash \Delta}{(\exists x)A(x), \Gamma \vdash \Delta}\ \exists_l$$

Let ρ be the sequence of descendents from $(\exists x)A(x)$. Let $t_1, \ldots, t_n$ be the terms introduced on ρ by $\exists_r$ rules. Let f be the Skolem symbol of the Skolemization of $(\exists x)A(x)$ in S. Replace proof of $(\exists x)A(x), \Gamma \vdash \Delta$ by $A(f(t_1, \ldots, t_n)), \Gamma \vdash \Delta$, modify descendents accordingly. □

Properties of Skolemized proofs

- Observe: Only ancestors of the end-sequent are modified
- Ancestors of cut-formulas are never ancestors of the end-sequent
- Obtain: In Skolemized proofs,
  1. all strong quantifier rules operate on cut-ancestors
  2. no strong quantifier rules operate on end-sequent ancestors
- Cut-free proofs are closed under substitution

Skolemizing cut?

$$\frac{\Gamma \vdash \Delta, (\exists x)P(x) \quad (\exists x)P(x), \Pi \vdash \Lambda}{\Gamma, \Pi \vdash \Delta, \Lambda} \; cut$$

a valid inference, but Skolemizing the cut-formulas yields

$$\frac{\Gamma \vdash \Delta, (\exists x)P(x) \quad P(c), \Pi \vdash \Lambda}{\Gamma, \Pi \vdash \Delta, \Lambda} \; \textit{not a cut}$$

Summary & Application

- Prefix vs. Structural: Structural wins
- Efficient structural proof Skolemization exists
- Application CERES

Proof De-Skolemization

- Original language $\mathcal{L}$, set of Skolem symbols $\mathcal{SK} = \{f_1, f_2, \ldots\}$
- J. Avigad 2003
  - Theory contains axioms $\forall \vec{x}, y(F_i(\vec{x}, y) \rightarrow F_i(\vec{x}, f_i(\vec{x})))$
  - Have proof in $\mathcal{L} \cup \mathcal{SK}$ of formula G in $\mathcal{L}$
  - Want proof in $\mathcal{L}$ of G
  - Result: If the theory allows coding of finite functions and we allow cuts, then this is possible with polynomial size increase

Proof De-Skolemization

- Original language $\mathcal{L}$, set of Skolem symbols $\mathcal{SK} = \{f_1, f_2, \ldots\}$
- J. Avigad 2003
- H. de Nivelle 2003
  - Have a resolution proof in $\mathcal{L} \cup \mathcal{SK}$
  - Introduce Skolem relations $\mathcal{SK}_R = \{R_{f_1}, R_{f_2}, \ldots\}$
  - Want a resolution proof in $\mathcal{L} \cup \mathcal{SK}_R$
  - Result: Possible with polynomial size increase

Proof De-Skolemization

- Original language $\mathcal{L}$, set of Skolem symbols $\mathcal{SK} = \{f_1, f_2, \ldots\}$
- J. Avigad 2003
- H. de Nivelle 2003
- Our version:

Problem (Proof De-skolemization)

Input: *Sequent S, cut-free* **LK***-proof of* $\mathrm{ssk}(S)$
Output: *cut-free* **LK***-proof of S*

Upper bound

Definition

$\mathcal{QMON}$ is the class of **LK**$\bot$-proofs π such that

1. the end-sequent of π is a QM-sequent and
2. all cut-formulas are monotone.

$\mathcal{QMON}^*$ is the class of right-normal $\mathcal{QMON}$-proofs.

Proposition (M. Baaz, A. Leitsch 1999)

Let $\pi \in \mathcal{QMON}^*$ be a contraction-normalized cut-free proof of a sequent S containing weak quantifiers only. Let S' be any sequent s.t. S is the Skolemization of S'. Then there exists a cut-free proof π' of S' with $l(\pi') \leq (\mathrm{quocc}(S') + 1)l(\pi)$, where quocc denotes the number of quantifier-occurrences.

Upper bound

Theorem

Let S be a closed sequent, and let π be a proof of $\mathrm{ssk}(S)$. Then there exists a proof φ of S such that $l(\varphi) < 3^{(\mathrm{quocc}(\mathrm{ssk}(S))+1)l(\pi)+1}$.

Upper bound

Proof sketch.

Skolem terms of the form $f(t_1, \ldots, t_n)$: f-Skolem terms.
Skolem terms not containing bound variables: free Skolem terms.
Idea: Eliminate Skolem terms one-by-one.
End-sequent of π: $\Gamma \vdash \Delta, G[C(t)]$. t is f-Skolem term.
Want proof of $\Gamma \vdash \Delta, G[(\exists y)C(y)]$.
If there are no free f-Skolem terms in ψ, then all ancestors of $C(t)$ are introduced by weakening. Modify weakenings to get proof of $\Gamma \vdash \Delta, G[(\exists y)C(y)]$.

Upper bound

Proof sketch.

Assume there is a free f-Skolem term t' that is maximal w.r.t. term inclusion. Construct proof of either

$$\Gamma \vdash \Delta, G[(\exists y)C(y)],$$

or

$$\Gamma \vdash \Delta, G[C(t)], G[(\exists y)C(y)]$$

depending on whether there is only one free f-Skolem term or more.

Upper bound

Proof sketch.

Goal: $\Gamma \vdash \Delta, G[C(t)], G[(\exists y)C(y)]$

First step: construct proof of $C(t')\sigma, \Gamma \vdash \Delta, G[C(t)]$ by projection method

Idea: Apply rules introducing the connectives from C, then if C contains t', do not apply rules introducing connectives from G. σ is the substitution induced by the $\exists_r$ we do not apply.

Length of resulting proof $\leq l(\pi)$.

Upper bound

Proof sketch.

Goal: $\Gamma \vdash \Delta, G[C(t)], G[(\exists y)C(y)]$

Have: $C(t')\sigma, \Gamma \vdash \Delta, G[C(t)]$

Replace t' by a new variable α. By assumption $\Gamma, \Delta, G[C(t)]$ closed, and by construction all occurrences of t' indicated in $C(t')$. Hence we can apply $\exists_l$ and get a proof of

$$(\exists y)C(y)\sigma, \Gamma \vdash \Delta, G[C(t)]$$

which has length $\leq l(\pi) + 1$.

To do: introduce connectives from G.

Upper bound

Proof sketch.

Goal: $\Gamma \vdash \Delta, G[C(t)], G[(\exists y)C(y)]$

Have: $(\exists y)C(y)\sigma, \Gamma \vdash \Delta, G[C(t)]$

Most complicated part of proof. We know from π which rules to apply to get $G[(\exists y)C(y)]$. The complication arises from applying binary rules which introduce new material through the context. In particular, t' may be re-introduced in this way. It turns out that all of these occurrences are eliminated by $\exists_r$ rules, so after constructing the desired proof, we can again replace t' by a new variable to completely eliminate it from the proof.

Upper bound

Proof sketch.

Have: $\Gamma \vdash \Delta, G[C(t)], G[(\exists y) C(y)]$

In addition to the proof obtained by the projection method, we apply at most $l(\pi)$ many rules. In the end, we have to apply at most $l(\pi)$ many contraction rules due to duplications due to context formulas. Hence our proof has length $\leq 3l(\pi)$. We have eliminated a free f-Skolem term, and there are at most $n = \mathrm{quocc}(\mathrm{ssk}(S))l(\pi)$ many such terms. After repeating the construction, we again have to insert at most n contractions and have obtained a proof φ of S. Finally we get $l(\varphi) < 3^{(\mathrm{quocc}(\mathrm{ssk}(S))+1)l(\pi)+1}$. □

Lower bound

- Calculus $\mathbf{LK}_p$
- Weakening restricted to be directly below axioms

$$\frac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, A \vee B} \vee_r^c$$

$$\frac{A, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \neg A} \neg_r \qquad \frac{\Gamma \vdash \Delta, A}{\neg A, \Gamma \vdash \Delta} \neg_l$$

$$\frac{\Gamma \vdash \Delta, (\exists x)A, A[x \leftarrow t]}{\Gamma \vdash \Delta, (\exists x)A} \exists_r^c \qquad \frac{A[x \leftarrow \alpha], \Gamma \vdash \Delta}{(\exists x)A, \Gamma \vdash \Delta} \exists_l$$

$$\frac{A, \Gamma, \Pi \vdash \Delta, \Lambda \quad B, \Gamma', \Pi \vdash \Delta', \Lambda}{A \vee B, \Gamma, \Gamma', \Pi \vdash \Delta, \Delta', \Lambda} \vee_l^c$$

$$\frac{\Gamma \vdash \Delta}{\Pi, \Gamma \vdash \Delta, \Lambda} w_*$$

Lower bound

- Calculus $\mathbf{LK}_p$
- Weakening restricted to be directly below axioms
- Polynomially equivalent to the usual formulations of **LK**

Lower bound

Theorem

There exists a sequence of sequents (S_N) *such that*

1. *for all* $\mathbf{LK}_p$*-proofs* π *of* S_N, $l(\pi) \geq 2^N + c$ *for some constant* c, *and*
2. *there exists an* $\mathbf{LK}_p$*-proof* π_{ssk} *of* $\mathrm{ssk}(S_N)$ *such that* $l(\pi_{\mathrm{ssk}}) \leq k * N + c$ *for some constants* c, k.

Lower bound

Proof sketch.

Take S_N to be $\vdash R_N$, where

$$\begin{aligned} R_0 &= \quad G_0 \rightarrow G_0 \\ R_n &= \quad ((\exists x_n)P_n(x_n) \vee G_n) \rightarrow (\exists y_n)((P_n(y_n) \vee G_n) \wedge R_{n-1}) \end{aligned}$$

Idea: Consider arbitrary $\mathbf{LK}_p$-proof π of S_N. As $\mathbf{LK}_p$ is rather "deterministic", there are not many possible ways to apply its rules.

Lower bound

1. Only $\rightarrow_r$ applicable

$$\frac{(\exists x_N)P_N(x_N) \vee G_N \vdash (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})}{\vdash ((\exists x_N)P_N(x_N) \vee G_N) \rightarrow (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})} \rightarrow_r$$

Lower bound

1. Only $\rightarrow_r$ applicable
2. $\exists_r$ not applicable (countermodel!), must apply $\vee_l^c$.
   Possibilities for π_1, π_2:
   2.1 $(\exists x_N)P_N(x_N) \vdash$ for π_1: $\lightning$
   2.2 $G_N \vdash$ for π_2: $\lightning$

$$\dfrac{\dfrac{\pi_1 \qquad \pi_2}{(\exists x_N)P_N(x_N) \vee G_N \vdash (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})}\ \vee_l^c}{((\exists x_N)P_N(x_N) \vee G_N) \rightarrow (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})}\ \rightarrow_r$$

Lower bound

1. Only $\rightarrow_r$ applicable
2. $\exists_r$ not applicable (countermodel!), must apply $\vee_l^c$. Possibilities for π_1, π_2:
   2.1 $(\exists x_N)P_N(x_N) \vdash$ for π_1: $\lightning$
   2.2 $G_N \vdash$ for π_2: $\lightning$
   2.3 $(\exists x_N)P_N(x_N) \vdash (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})$ for π_1, $G_N \vdash (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})$ for π_2.

$$\dfrac{\dfrac{\pi_1 \qquad \pi_2}{(\exists x_N)P_N(x_N) \vee G_N \vdash (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})}\ \vee_l^c}{((\exists x_N)P_N(x_N) \vee G_N) \rightarrow (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})}\ \rightarrow_r$$

Lower bound

1. Again $\exists_r$ is not applicable, must apply $\exists_l$.

$$\frac{P_N(\alpha_N) \vdash (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})}{(\exists x_N)P_N(x_N) \vdash (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})}\ \exists_l$$

Lower bound

1. Again $\exists_r$ is not applicable, must apply $\exists_l$.
2. Only applicable rule is $\exists_r$, instantiating some term t.

$$\dfrac{\dfrac{P_N(\alpha_N) \vdash (P_N(t) \vee G_N) \wedge R_{N-1}, (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})}{P_N(\alpha_N) \vdash (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})}\ \exists_r^c}{(\exists x_N) P_N(x_N) \vdash (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})}\ \exists_l$$

Lower bound

1. Again $\exists_r$ is not applicable, must apply $\exists_l$.
2. Only applicable rule is $\exists_r$, instantiating some term t.
3. Two $\exists_r$ never need to be applied consecutively on the same formula. We have to apply $\wedge_r$.

$$\dfrac{\dfrac{\dfrac{\pi_1^1 \qquad \pi_1^2}{P_N(\alpha_N) \vdash (P_N(t) \vee G_N) \wedge R_{N-1}, (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})}\ \wedge_r}{P_N(\alpha_N) \vdash (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})}\ \exists_r^c}{(\exists x_N) P_N(x_N) \vdash (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})}\ \exists_l$$

Lower bound

1. Again $\exists_r$ is not applicable, must apply $\exists_l$.
2. Only applicable rule is $\exists_r$, instantiating some term t.
3. Two $\exists_r$ never need to be applied consecutively on the same formula. We have to apply $\wedge_r$.
4. Right subproof must be $\vdash R_{N-1}$. Otherwise we have to prove either
   4.1 $\vdash (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1}), R_{N-1}$. Neither P_N nor G_N occur in R_{N-1}, so a proof of this is at least as long as the shortest proof of $\vdash R_{N-1}$.
   4.2 $P_N(\alpha_N) \vdash R_{N-1}$. P_N does not occur in R_{N-1}, so a proof of this is at least as long as the shortest proof of $\vdash R_{N-1}$.
   4.3 $P_N(\alpha_N) \vdash (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1}), R_{N-1}$. P_N does not occur in R_{N-1}, so we have to prove $P_N(\alpha_N) \vdash (\exists y_N)((P_N(y_N) \vee G_N) \wedge R_{N-1})$, but then the shortest proof must contain itself: $\lightning$.

Lower bound

The argument for π_2 is similar. We obtain that π_1, π_2 must both contain proofs of $\vdash R_{N-1}$, hence by induction we get that $l(\pi) \geq 2^N + c$ for some constant c.

Lower bound

Now we give the short $\mathbf{LK}_p$-proof of $\mathrm{ssk}(S_N)$.
Set $s_n^N = f_n(y_N, y_{N-1}, \ldots, y_{n+1})$, then $\mathrm{ssk}(S_N)$ is $\vdash K_N^N$ where

$$\begin{aligned} K_0^N = & \quad G_0 \to G_0 \\ K_n^N = & \quad (P_n(s_n^N) \vee G_n) \to (\exists y_n)((P_n(y_n) \vee G_n) \wedge K_{n-1}^N) \end{aligned}$$

Let σ be any substitution, then we give a proof of $\vdash K_n^N \sigma$.

Lower bound

$$\dfrac{\dfrac{\dfrac{\dfrac{\dots}{P_n(s_n^N\sigma) \vee G_n \vdash P_n(s_n^N\sigma) \vee G_n}\ \vee_r^c \qquad \vdash K_{n-1}^N\sigma\{y_n \leftarrow s_n^N\sigma\}}{P_n(s_n^N\sigma) \vee G_n \vdash (P_n(s_n^N\sigma) \vee G_n) \wedge K_{n-1}^N\sigma\{y_n \leftarrow s_n^N\sigma\}}\ \wedge_r^c}{P_n(s_n^N\sigma) \vee G_n \vdash (\exists y_n)((P_n(y_n) \vee G_n) \wedge K_{n-1}^N\sigma)}\ \exists_r^c + w_r}{\vdash (P_n(s_n^N\sigma) \vee G_n) \rightarrow (\exists y_n)((P_n(y_n) \vee G_n) \wedge K_{n-1}^N\sigma)}\ \rightarrow_r$$

By induction hypothesis, we have a proof of $\vdash K_{n-1}^N\sigma\{y_n \leftarrow s_n^N\sigma\}$ of length $\leq k*(n-1)+c$, so this proof has length $\leq k*n+c$. □

Summary & Application

- Efficient de-Skolemization impossible in *tree-like* **LK**
- Application CERES: Elimination of single cuts

Future Work

- Complexity of de-Skolemization in DAG-like **LK**
- Complexity of de-Skolemization w.r.t. CERES
- Does the de-Skolemization proof work with Andrews Skolemization?