Towards CERES in Higher-Order Logic

Daniel Weller

February 17, 2010

Motivation

- Cut-elimination (Gentzen 1935) makes implicit information in proofs explicit:
  - Cut-free proofs have the subformula property.
- Cut-elimination is highly non-confluent (Baaz, Hetzl 2010)
  - Proofs may give rise to non-elementarily many cut-free proofs *with significantly different Herbrand disjunctions*.

Motivation

- Interesting application: Mathematical proofs, i.e. proof mining, extract information from (classical) mathematical proofs.
- Cut-elimination corresponds to the removal of lemmas.
- Different technique: Functional interpretation (see e.g. Kohlenbach 2008).

Cut-elimination methods in FOL

- Reductive methods: Gentzen 1935, Tait 1968.
- Based on proof rewrite rules.
- Cut-elimination by resolution (CERES): Baaz, Leitsch 2000.
- Use resolution to find different cut-free proofs.

Some properties of CERES

- CERES simulates the reductive methods up to an exponential.

Theorem (Baaz, Leitsch 2006)

*Let φ be an **LK**-derivation and ψ be an ACNF of φ under a cut reduction relation $>_{\mathcal{R}}$ based on $\mathcal{R}$. Then there exists an ACNF χ of φ under CERES such that*

$$l(\chi) \leq l(\varphi) * l(\psi) * 2^{2*l(\psi)} + 2.$$

Some properties of CERES

- CERES simulates the reductive methods up to an exponential.
- There is a non-elementary speed up of CERES over the reductive methods.

Theorem (Baaz, Leitsch 2000)

There exists a sequence of **LK***-proofs* $(\psi_n)_{n\in\mathbb{N}}$ *such that*

1. *The Gentzen method produces proof trees with non-elementarily many nodes on* ψ_n.
2. *CERES constructs a cut-free proof out of* ψ_n *in exponentially many steps.*

Some properties of CERES

- CERES simulates the reductive methods up to an exponential.
- There is a non-elementary speed up of CERES over the reductive methods.
- CERES has been used to prove fast cut-elimination for classes for which the reductive methods cannot be used. (Baaz, Leitsch 2010?)

Applying CERES

- First idea: Using powerful resolution provers, apply cut-elimination completely automated.
- Partial success: Works fine on simple proofs.
- Current Implementation: ANSI C++. (Work-in-progress implementation: Scala.)

Applying CERES — examples

Example (The tape proof)

- A version of the pigeon hole principle: The "tape proof" due to C. Urban.
- On a tape with infinitely many cells, each labelled either 0 or 1, there are two distinct cells with the same label.
- Uses a classical lemma: Either infinitely many cells are labelled 0, or infinitely many cells are labelled 1.

Analysis in Baaz, Hetzl, Leitsch, Richter, Spohr 2006.

Applying CERES — examples

Example (The lattice proof)

There are different equivalent formulations of the notion of lattice:

1. $\langle S, \cap, \cup \rangle$ such that $\cup$ and $\cap$ are commutative, associative, idempotent and "inverse".
2. $\langle S, \cap, \cup \rangle$ such that $\cup$ and $\cap$ are commutative, associative, idempotent and two "absorption laws" hold.
3. A partially ordered set $\langle S, \leq \rangle$ such that $\cap$ is the greatest lower bound and $\cup$ is the least upper bound.

One proves (1) $\rightarrow$ (2) by proving (1) $\rightarrow$ (3) and (3) $\rightarrow$ (2).

Analysis in Hetzl, Leitsch, Weller, Woltzenlogel Paleo 2008.

Applying CERES

- First-order theorem provers used in the experiments: Otter, Prover9.
- Problems with more complicated proofs:
  - Induction.
  - Theorem provers fail to find refutation automatically.

Fürstenberg's proof of the infinitude of primes

Example (Fürstenberg's proof)

- Proof of the infinitude of primes by topological means.
- Topology is induced by arithmetic progressions over the integers.

Analysis in Baaz, Hetzl, Leitsch, Richter, Spohr 2008.

Fürstenberg's proof

- Proof by contradiction: Assume the set of primes has cardinality k, derive a contradiction.
- Induction is used to establish this.
- In the experiment, induction is treated via *schematization*.

Schematization

- Advantages:
  - Proof in Robinson arithmetic.
  - The problem of cut-elimination is partitioned into cut-elimination for $k = 0, 1, 2, \ldots$.
  - Induction is moved to the meta-level.
- Disadvantages:
  - No formal basis (yet), therefore:
  - The general form of the CERES datastructures for k arbitrary has to be determined empirically.

Fürstenberg's proof

- Prover9 finds refutations for $k = 0, 1, 2$.
- It was not clear how to generalize the refutations. (IS THIS TRUE?)
- Manually, a (inductively defined) refutation was found for all k.
- In it, a construction central to Euclid's original argument appears: $p_1 * \cdots * p_k + 1$.

Post-experiment

- Completely automated cut-elimination seems unrealistic: Instead, apply semi-automatically.
- Human effort: Try to understand and refute the *characteristic clause set*.
- Make easier by moving to more expressive formalism: HOL.
- Allows to move induction from meta- to object-level.

CERES — Method overview

1. Input proof in sequent calculus format.
2. Move to proof format which is more flexible with respect to structural manipulations ("sequents + skolemization").
3. Compute characteristic clause set.
4. Refute the clause set.
5. From the refutation, build a proof with at most atomic cuts.

CERES — Method overview

1. Input proof in sequent calculus format.
2. Move to proof format which is more flexible with respect to structural manipulations ("sequents + skolemization").
3. Compute characteristic clause set.
4. Apply subsumption and other pruning techniques to reduce its size.
5. Refute the clause set.
6. From the refutation, build a proof with at most atomic cuts.

CERES — central constructions

- Input proof π of S.
- *Characteristic clause set* $\mathrm{CL}(\pi)$.
- For every $C \in \mathrm{CL}(\pi)$, a proof $\pi(C)$ of $C \circ S$ (*proof projection*).

The characteristic clause set $\mathrm{CL}(\pi)$

- Intuition: Collect material from the cuts. *How* depends on the shape of π.
- For every inference ρ in π, $\mathrm{CL}_\rho(\pi)$ is defined.

The characteristic clause set $\mathrm{CL}(\pi)$

- For axioms A, $\mathrm{CL}_\rho(\pi) = \{c(A)\}$ where $c(A)$ is the sub-sequent of A consisting of the cut-ancestors,
- For unary rules with premise σ, $\mathrm{CL}_\rho(\pi) = \mathrm{CL}_\sigma(\pi)$.
- For binary rules with premises σ_1, σ_2:
  - If it operates on cut ancestors, $\mathrm{CL}_\rho(\pi) = \mathrm{CL}_{\sigma_1}(\pi) \cup \mathrm{CL}_{\sigma_2}(\pi)$.
  - Otherwise, $\mathrm{CL}_\rho(\pi) = \mathrm{CL}_{\sigma_1}(\pi) \times \mathrm{CL}_{\sigma_2}(\pi)$.

The characteristic clause set $\mathrm{CL}(\pi)$

Theorem

There exists a refutation of $\mathrm{CL}(\pi)$.

Proof sketch.

For every inference ρ with conclusion S in π, we construct a proof of $c(S)$. □

The characteristic clause set $\mathrm{CL}(\pi)$

The construction of

- the characteristic clause set $\mathrm{CL}(\pi)$ and
- its refutation in the sequent calculus

both go through in HOL.

Constructing an ACNF — in FOL

Theorem

There exists a resolution refutation of $\mathrm{CL}(\pi)$.

Proof.

By soundness of the sequent calculus and completeness of the resolution calculus. □

Constructing an ACNF — in FOL

- π is a proof of S.
- We have a resolution refutation γ of $\mathrm{CL}(\pi)$.
- We want: A proof of S with at most atomic cuts.
- Intuition: *Ground* resolution refutation is a sequent calculus refutation with at most atomic cuts!
- Combine with *proof projections*.

Constructing proof projections — in FOL

- We construct proofs of $C \circ S$.
- Inductive construction analogous to that of $\mathrm{CL}(\pi)$.
- Intuition: We take π, but apply only rules that operate on end-sequent ancestors.

Constructing proof projections — in FOL

- Crucial case: strong quantifier rules

$$\frac{\Gamma \vdash \Delta, F(\alpha)}{\Gamma \vdash \Delta, (\forall x)F(x)}\ \forall_r$$

  where α must not occur in $\Gamma, \Delta, F(x)$.
- If a clause contains α, we cannot apply the rule!
- Solution: *Proof skolemization*.

CERES — Method overview

1. Input proof in sequent calculus format.
2. Move to proof format which is more flexible with respect to structural manipulations (“sequents + skolemization”).
3. Compute characteristic clause set.
4. Refute the clause set.
5. From the refutation, build a proof with at most atomic cuts.

Proof skolemization

- Roughly, skolemization sk removes quantifiers $(\forall x)$ and replaces x by a skolem term $f(y_1, \ldots, y_n)$ where f is a fresh function symbol.
- Crucial property of proofs of skolemized sequents: "by the subformula property", no strong quantifier rules operate on end-sequent ancestors.

Proposition

There exists a proof of S $\iff$ there exists a proof of $\mathrm{sk}(S)$.

Proof skolemization

- In HOL, proof skolemization is possible, but does not yield the desired property:
- The subformula property is modulo "formula substitution", not modulo "term substitution"!
- Hence quantifiers may not only be introduced in the end-sequent.

Comprehension

$$\frac{\overline{\mathbf{F}\mathbf{T}}, \Gamma \vdash \Delta}{\forall \mathbf{F}, \Gamma \vdash \Delta} \; \forall : l$$

where **T** is a HOL term (and hence may contain quantifiers).

Approach: Modify the sequent calculus

- Define cut-free sequent calculus $\mathbf{LK}_{\mathrm{sk}}$ that introduces strong quantifiers from skolem terms.
- Replace "eigenfunction" condition by global conditions.
- Similar to how strong quantifiers are treated in skolem expansion trees (Miller 1983).
- Hope: In sequent format, structural transformations necessary for CERES will be easier than with more compact formalisms.

$\mathbf{LK}_{\mathrm{sk}}$ — crucial rules

Labelled formulas $\langle\cdot\rangle^{\ell}$ where ℓ is a set of terms.

$$\frac{\Gamma \vdash \Delta, \left\langle \overline{\mathbf{F}(\mathbf{f}\mathbf{S}_1 \ldots \mathbf{S}_m)} \right\rangle^{\ell}}{\Gamma \vdash \Delta, \langle \forall_\alpha \mathbf{F} \rangle^{\ell}} \ \forall^{sk}: r \qquad \frac{\langle \overline{\mathbf{FT}} \rangle^{\ell, \mathbf{T}}, \Gamma \vdash \Delta}{\langle \forall_\alpha \mathbf{F} \rangle^{\ell}, \Gamma \vdash \Delta} \ \forall^{sk}: l$$

$\mathbf{f}$ is a Skolem function, $\ell \subseteq \{\mathbf{S}_1, \ldots, \mathbf{S}_m\}$.

Regularity

- Intuition for usual quantifier rules: Different inferences should use different variables (*regularity*).
- There are proofs which are not regular: Eigenvariable condition suffices for soundness.
- But: there are *transformations* which require regularity to fulfill the eigenvariable condition.
- In $\mathbf{LK}_{\mathrm{sk}}$, we will use analogies to regularity to ensure soundness.

Weak regularity

- Introduce notion of *weak regularity*.
- Intuition: If objects have the same name, then they are used in the same way.

Definition

An $\mathbf{LK}_{\mathrm{sk}}$-tree is weakly regular if for every two strong quantifier inferences ρ_1, ρ_2: If ρ_1, ρ_2 have the same skolem term, then they are *homomorphic*.

Roughly, two inferences are homomorphic if on the paths starting at their auxiliary formulas, the same inferences are applied, and they are joined in a contraction.

Soundness and completeness

Theorem (Completeness)

For every **LK***-proof of* S*, there exists a weakly regular* $\mathbf{LK}_{\mathrm{skc}}$*-tree of* S*.*

Proof sketch.

We replace eigenvariables by appropriate skolem terms. □

Note: We can even construct an $\mathbf{LK}_{\mathrm{skc}}$-tree where the skolem terms of strong quantifier inferences are pairwise different. In practice, we will want to exploit weak-regularity already here, to reduce the number of different Skolem functions.

Soundness and completeness

Theorem (Soundness)

For every weakly regular $\mathbf{LK}_{\mathrm{sk}}$*-proof* π *of* S*, there exists an* **LK***-proof of* S*.*

Proof sketch.

By structural manipulation (rule permutations and pruning), π is brought into a form where an "eigenterm condition" holds. Then Skolem terms are replaced by eigenvariables. □

CERES — Method overview

1. Input proof in sequent calculus format.
2. Move to proof format which is more flexible with respect to structural manipulations (“sequents + skolemization”).
3. Compute characteristic clause set.
4. Refute the clause set.
5. From the refutation, build a proof with at most atomic cuts.

Constructing proof projections — in HOL

- For all $C \in \mathrm{CL}(\pi)$ we can now construct appropriate $\mathbf{LK}_{\mathrm{sk}}$-trees of $S \circ C$.

Proposition

Let π be a regular $\mathbf{LK}_{\mathrm{skc}}$-proof of S. For every $C \in \mathrm{CL}(\pi)$, there exists a regular $\mathbf{LK}_{\mathrm{sk}}$-tree $\pi(C) \in \mathcal{P}(\pi)$ of $S \circ C$ such that

1. *$\pi(C)$ is S-balanced, and*
2. *if ω is a formula occurrence in C in the end-sequent of $\pi(C)$ with label l then ω has exactly one axiom partner μ, and μ also has label l, and*
3. *$l(\pi(C)) \leq l(\pi)$.*

Moreover, for all $C_1, C_2 \in \mathrm{CL}(\pi)$, $\pi(C_1)$, $\pi(C_2)$ are Skolem parallel with respect to S.

CERES — Method overview

1. Input proof in sequent calculus format.
2. Move to proof format which is more flexible with respect to structural manipulations ("sequents + skolemization").
3. Compute characteristic clause set.
4. Refute the clause set.
5. From the refutation, build a proof with at most atomic cuts.

Resolution calculus $\mathcal{R}$

$$\frac{\Gamma \vdash \Delta, \langle \neg \mathbf{A} \rangle^{\ell}}{\langle \mathbf{A} \rangle^{\ell}, \Gamma \vdash \Delta} \neg^{T} \qquad \frac{\langle \neg \mathbf{A} \rangle^{\ell}, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \langle \mathbf{A} \rangle^{\ell}} \neg^{F} \qquad \frac{\Gamma \vdash \Delta, \langle \mathbf{A} \vee \mathbf{B} \rangle^{\ell}}{\Gamma \vdash \Delta, \langle \mathbf{A} \rangle^{\ell}, \langle \mathbf{B} \rangle^{\ell}} \vee^{T}$$

$$\frac{\langle \mathbf{A} \vee \mathbf{B} \rangle^{\ell}, \Gamma \vdash \Delta}{\langle \mathbf{A} \rangle^{\ell}, \Gamma \vdash \Delta} \vee^{F}_{l} \qquad \frac{\langle \mathbf{A} \vee \mathbf{B} \rangle^{\ell}, \Gamma \vdash \Delta}{\langle \mathbf{B} \rangle^{\ell}, \Gamma \vdash \Delta} \vee^{F}_{r} \qquad \frac{\Gamma \vdash \Delta, \langle \forall_{\alpha} \mathbf{A} \rangle^{\ell}}{\Gamma \vdash \Delta, \langle \mathbf{A}\mathbf{X} \rangle^{\ell, \mathbf{X}}} \forall^{T}$$

$$\frac{\langle \forall_{\alpha} \mathbf{A} \rangle^{\ell}, \Gamma \vdash \Delta}{\langle \mathbf{A}\mathbf{S} \rangle^{\ell}, \Gamma \vdash \Delta} \forall^{F} \qquad \frac{S}{S[\mathbf{X} \leftarrow \mathbf{T}]} \mathrm{Sub}$$

$$\frac{\langle \mathbf{A} \rangle^{\ell_1}, \langle \mathbf{A} \rangle^{\ell_2}, \Gamma \vdash \Delta}{\langle \mathbf{A} \rangle^{\ell_1, \ell_2}, \Gamma \vdash \Delta} \mathrm{Sim}^{F} \qquad \frac{\Gamma \vdash \Delta, \langle \mathbf{A} \rangle^{\ell_1}, \langle \mathbf{A} \rangle^{\ell_2}}{\Gamma \vdash \Delta, \langle \mathbf{A} \rangle^{\ell_1, \ell_2}} \mathrm{Sim}^{T}$$

$$\frac{\Gamma \vdash \Delta, \langle \mathbf{A} \rangle^{\ell_1} \quad \langle \mathbf{A} \rangle^{\ell_2}, \Pi \vdash \Lambda}{\Gamma, \Pi \vdash \Delta, \Lambda} \mathrm{Cut}$$

Resolution calculus $\mathcal{R}$

- Similar to Andrews' higher-order resolution calculus.
- Just like Andrews, we require: Every strong quantifier rule has a unique Skolem function.
- Unlike Andrews, we use resolution trees instead of DAGs!
- Completeness?

Resolution calculus $\mathcal{R}$ vs. FOL resolution

$$\frac{\Gamma \vdash \Delta, \langle \neg \mathbf{A} \rangle^{\ell}}{\langle \mathbf{A} \rangle^{\ell}, \Gamma \vdash \Delta} \neg^{T} \qquad \frac{\langle \neg \mathbf{A} \rangle^{\ell}, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \langle \mathbf{A} \rangle^{\ell}} \neg^{F} \qquad \frac{\Gamma \vdash \Delta, \langle \mathbf{A} \vee \mathbf{B} \rangle^{\ell}}{\Gamma \vdash \Delta, \langle \mathbf{A} \rangle^{\ell}, \langle \mathbf{B} \rangle^{\ell}} \vee^{T}$$

$$\frac{\langle \mathbf{A} \vee \mathbf{B} \rangle^{\ell}, \Gamma \vdash \Delta}{\langle \mathbf{A} \rangle^{\ell}, \Gamma \vdash \Delta} \vee^{F}_{l} \qquad \frac{\langle \mathbf{A} \vee \mathbf{B} \rangle^{\ell}, \Gamma \vdash \Delta}{\langle \mathbf{B} \rangle^{\ell}, \Gamma \vdash \Delta} \vee^{F}_{r} \qquad \frac{\Gamma \vdash \Delta, \langle \forall_{\alpha} \mathbf{A} \rangle^{\ell}}{\Gamma \vdash \Delta, \langle \mathbf{A} \mathbf{X} \rangle^{\ell, \mathbf{X}}} \forall^{T}$$

$$\frac{\langle \forall_{\alpha} \mathbf{A} \rangle^{\ell}, \Gamma \vdash \Delta}{\langle \mathbf{A} \mathbf{S} \rangle^{\ell}, \Gamma \vdash \Delta} \forall^{F} \qquad \frac{S}{S[\mathbf{X} \leftarrow \mathbf{T}]} \mathrm{Sub}$$

$$\frac{\langle \mathbf{A} \rangle^{\ell_1}, \langle \mathbf{A} \rangle^{\ell_2}, \Gamma \vdash \Delta}{\langle \mathbf{A} \rangle^{\ell_1, \ell_2}, \Gamma \vdash \Delta} \mathrm{Sim}^{F} \qquad \frac{\Gamma \vdash \Delta, \langle \mathbf{A} \rangle^{\ell_1}, \langle \mathbf{A} \rangle^{\ell_2}}{\Gamma \vdash \Delta, \langle \mathbf{A} \rangle^{\ell_1, \ell_2}} \mathrm{Sim}^{T}$$

$$\frac{\Gamma \vdash \Delta, \langle \mathbf{A} \rangle^{\ell_1} \qquad \langle \mathbf{A} \rangle^{\ell_2}, \Pi \vdash \Lambda}{\Gamma, \Pi \vdash \Delta, \Lambda} \mathrm{Cut}$$

Putting things together

- In FOL, a ground resolution refutation is essentially an **LK**-refutation.
- In HOL, things are more complicated due to the CNF rules.
- To combine the $\mathcal{R}$-refutation and the projections, we combine the rules to form $\mathbf{LK}_{\mathrm{sk}}$-$\mathcal{R}$-trees.

Putting things together

- The $\mathbf{LK}_{\mathrm{sk}}$-projections and the $\mathcal{R}$-refutation of $\mathrm{CL}(\pi)$ are plugged together to form an $\mathbf{LK}_{\mathrm{sk}}$-$\mathcal{R}$-tree of the end-sequent of π (*CERES-proof*).
- Objective: Convert this $\mathbf{LK}_{\mathrm{sk}}$-$\mathcal{R}$-tree into a weakly regular $\mathbf{LK}_{\mathrm{sk}}$-tree.
- By the soundness theorem for $\mathbf{LK}_{\mathrm{sk}}$, we can then obtain a cut-free **LK**-proof.

From $\mathbf{LK}_{\mathrm{sk}}$-$\mathcal{R}$ to $\mathbf{LK}_{\mathrm{sk}}$

Lemma

Let π be a CERES*-proof of S. Then there exists a pre-regular, cut-free $\mathbf{LK}_{\mathrm{sk}}$-$\mathcal{R}$-tree ψ of S.*

Proof sketch.

We eliminate the (atomic!) cuts (all $\mathcal{R}$-inferences operate on cut-ancestors).

1. Shift up the $\mathcal{R}$-inferences.
2. At the leaves:
   - Convert CNF rules into logical **LK**-rules,
   - eliminate cuts,
   - absorb Sub inferences.

□

Permuting up $\mathcal{R}$ inference

- Inferences are duplicated when shifted over contractions (former $\mathrm{Sim}^{T}, \mathrm{Sim}^{F}$ inferences).
- Crucial case: Duplication of $\forall^{F}$ inferences: they are not homomorphic!
- Introduce another notion of regularity (later in this talk).

Converting $\mathcal{R}$ inferences

$$\frac{\langle \mathbf{A} \vee \mathbf{B} \rangle^{\ell} \vdash \langle \mathbf{A} \vee \mathbf{B} \rangle^{\ell}}{\langle \mathbf{A} \vee \mathbf{B} \rangle^{\ell} \vdash \langle \mathbf{A} \rangle^{\ell}, \langle \mathbf{B} \rangle^{\ell}} \vee^{T} \quad \rightsquigarrow \quad \frac{\langle \mathbf{A} \rangle^{\ell} \vdash \langle \mathbf{A} \rangle^{\ell} \quad \langle \mathbf{B} \rangle^{\ell} \vdash \langle \mathbf{B} \rangle^{\ell}}{\langle \mathbf{A} \vee \mathbf{B} \rangle^{\ell} \vdash \langle \mathbf{A} \rangle^{\ell}, \langle \mathbf{B} \rangle^{\ell}} \vee : l$$

$$\frac{\langle \forall_{\alpha} \mathbf{A} \rangle^{\ell} \vdash \langle \forall_{\alpha} \mathbf{A} \rangle^{\ell}}{\langle \forall_{\alpha} \mathbf{A} \rangle^{\ell} \vdash \langle \overline{\mathbf{AX}} \rangle^{\ell, \mathbf{X}}} \forall^{T} \quad \rightsquigarrow \quad \frac{\langle \overline{\mathbf{AX}} \rangle^{\ell, \mathbf{X}} \vdash \langle \overline{\mathbf{AX}} \rangle^{\ell, \mathbf{X}}}{\langle \forall_{\alpha} \mathbf{A} \rangle^{\ell} \vdash \langle \overline{\mathbf{AX}} \rangle^{\ell, \mathbf{X}}} \forall^{sk} : l$$

$$\frac{\langle \forall_{\alpha} \mathbf{A} \rangle^{\ell} \vdash \langle \forall_{\alpha} \mathbf{A} \rangle^{\ell}}{\langle \overline{\mathbf{AS}} \rangle^{\ell} \vdash \langle \forall_{\alpha} \mathbf{A} \rangle^{\ell}} \forall^{F} \quad \rightsquigarrow \quad \frac{\langle \overline{\mathbf{AS}} \rangle^{\ell} \vdash \langle \overline{\mathbf{AS}} \rangle^{\ell}}{\langle \overline{\mathbf{AS}} \rangle^{\ell} \vdash \langle \forall_{\alpha} \mathbf{A} \rangle^{\ell}} \forall^{sk} : r$$

Another notion of regularity

- Weak regularity: "If objects have the same name, then they are used in the same way."
- Now: "If objects have the same name, then they are either used in the same way, or not used together at all."
- *Weak+ regularity*.

Weak+ regularity

- Define a notion of connectedness of term occurrences via
  - The occurrence ancestor relation,
  - contractions, and
  - weak quantifier rules.

$$\frac{\langle \mathbf{A} \rangle^{\ell_1}, \langle \mathbf{A} \rangle^{\ell_2}, \Gamma \vdash \Delta}{\langle \mathbf{A} \rangle^{\ell_1,\ell_2}, \Gamma \vdash \Delta}\ \mathrm{Sim}^{F} \qquad \frac{\Gamma \vdash \Delta, \langle \forall_\alpha \mathbf{A} \rangle^{\ell}}{\Gamma \vdash \Delta, \langle \mathbf{AX} \rangle^{\ell,\mathbf{X}}}\ \forall^{T}$$

Weak+ regularity

- Roughly, weak+ regularity requires strong quantifier rules with the same Skolem term to either be
  - homorphic or
  - their Skolem function occurrences to be disconnected in the term connectedness graph.

Soundness

Theorem

Let π be a weakly+ regular, proper $\mathbf{LK}_{\mathrm{sk}}$*-tree of S. Then there exists a weakly-regular, proper* $\mathbf{LK}_{\mathrm{sk}}$*-tree of S.*

Proof sketch.

By renaming Skolem symbols modulo homomorphism equivalence classes. □

From $\mathbf{LK}_{\mathrm{sk}}$-$\mathcal{R}$ to $\mathbf{LK}_{\mathrm{sk}}$

Lemma

Let π be a CERES-proof of S. Then there exists a pre-regular, cut-free $\mathbf{LK}_{\mathrm{sk}}$-$\mathcal{R}$-tree ψ of S.

Proof sketch.

We eliminate the (atomic!) cuts (all $\mathcal{R}$-inferences operate on cut-ancestors).

1. Shift up the $\mathcal{R}$-inferences.
2. At the leaves:
   - Convert CNF rules into logical **LK**-rules,
   - eliminate cuts,
   - absorb Sub inferences.

□

Duplication of $\forall^F$ inferences

$$\dfrac{\dfrac{\langle \forall_\alpha \mathbf{A} \rangle^{\ell_1}, \langle \forall_\alpha \mathbf{A} \rangle^{\ell_2}, \Gamma \vdash \Delta}{\langle \forall_\alpha \mathbf{A} \rangle^{\ell_1,\ell_2}, \Gamma \vdash \Delta}\ \mathrm{Sim}^F}{\left\langle \overline{\mathbf{AS}} \right\rangle^{\ell_1,\ell_2}, \Gamma \vdash \Delta}\ \forall^F \quad \rightsquigarrow \quad \dfrac{\dfrac{\dfrac{\langle \forall_\alpha \mathbf{A} \rangle^{\ell_1}, \langle \forall_\alpha \mathbf{A} \rangle^{\ell_2}, \Gamma \vdash \Delta}{\left\langle \overline{\mathbf{AS}} \right\rangle^{\ell_1}, \langle \forall_\alpha \mathbf{A} \rangle^{\ell_2}, \Gamma \vdash \Delta}\ \forall^F}{\left\langle \overline{\mathbf{AS}} \right\rangle^{\ell_1}, \left\langle \overline{\mathbf{AS}} \right\rangle^{\ell_2}, \Gamma \vdash \Delta}\ \forall^F}{\left\langle \overline{\mathbf{AS}} \right\rangle^{\ell_1,\ell_2}, \Gamma \vdash \Delta}\ \mathrm{Sim}^F$$

- $\forall^F$ inferences not disconnected, but weakly disconnected: all connections go through a contraction!
- This property is preserved throughout the transformation.

From weakly disconnected to weakly+ regular

- All non-homomorphic strong quantifier inferences are weakly disconnected, and the $\mathbf{LK}_{\mathrm{sk}}$-tree is cut-free.
- $\rightsquigarrow$ we shift contraction inferences down: Now all such inferences are disconnected!
- Apply previous soundness theorems.

Completeness of CERES in HOL

- Method is not yet proven complete.
- We would like to have

Proposition

If there exists an **LK***-refutation of* $\mathrm{CL}(\pi)$*, then there exists an* $\mathcal{R}$*-refutation of* $\mathrm{CL}(\pi)$*.*

- Cannot directly use Andrews' completeness for V-complexes: our calculus has subtle differences:
  - Tree vs. DAG.
  - Labels vs. free variables.

Strategies for proving completeness

- Syntactically: transform Andrews' refutations into $\mathcal{R}$-refutations.
- Semantically: Give direct completeness proof of $\mathcal{R}$ w.r.t. V-complexes.

Implementing CERES for HOL

- As mentioned: we want to apply CERES to analyze proofs from mathematics.
- Old C++ implementation of CERES for FOL had several drawbacks:
  - The FO language was central to the implementation.
  - Hard-to-use reference-counting memory management.
  - Implementation of recursive algorithms with the visitor design pattern lead to lots of "boilerplate code".
- New implementation in *Scala*.

Implementing CERES for HOL

- Scala combines functional and object-oriented programming.
- Well suited for our purposes:
  - Efficiency not a priority.
  - Functional constructs allow easy implementation of algorithms.
  - Object orientation allows structuring of code in a natural way.
  - Built for HOL from the ground up.
- Scala compiles to Java bytcode: Platform independence, may re-use Java libraries.

State of implementation

- **LK**, $\mathbf{LK}_{\mathrm{skc}}$ and $\mathbf{LK}_{\mathrm{sk}}$ ✓
- Transformation from **LK** to $\mathbf{LK}_{\mathrm{skc}}$ ✓
- Construction of $\mathrm{CL}(\pi)$ ✓

Current experiment

- Formalization of Fürstenberg's proof of the infinitude of primes in second-order arithmetic (actually ACA_0).
- How does the induction behave on the object level?
- How does the modified subformula property affect the method in practice?

Future work

- Prove completeness of CERES.
- Check whether (skolem) expansion tree proofs can be extracted directly from $\mathbf{LK}_{\mathrm{sk}}$-$\mathcal{R}$-trees — implementation of soundness theorems can then be circumvented.