CERES and Fast Cut-Elimination

Mikheil Rukhaia

Institute of Computer Languages,
Vienna University of Technology.

Symposium on Language, Logic and Computation,
Kutaisi, Georgia.

September 30, 2011.

Introduction

Overview

- Cut-elimination is a proof transformation that removes all cut rules from a proof.
- The cut-elimination theorem was proved by G. Gentzen in 1934.
- For the systems, that have a cut-elimination theorem, it is easy to prove consistency.
- Cut-elimination is nonelementary in general, i.e. there is no elementary bound on the size of cut-free proof w.r.t the original one.

Sequent Calculus **LK**

- Sequent is an expression of the form $\Gamma \vdash \Delta$, where Γ and Δ are multisets of formulas.
- Rule is an inference of a lower sequent from an upper sequent.
- Derivation is a directed tree with nodes as sequences and edges as inferences.
- Proof of the sequence S is a derivation of S with axioms as leaf nodes.

Propositional rules

- $\wedge$ introduction:

$$\frac{A, \Gamma \vdash \Delta}{A \wedge B, \Gamma \vdash \Delta}\ \wedge: l1 \qquad \frac{B, \Gamma \vdash \Delta}{A \wedge B, \Gamma \vdash \Delta}\ \wedge: l2$$

$$\frac{\Gamma \vdash \Delta, A \qquad \Pi \vdash \Lambda, B}{\Gamma, \Pi \vdash \Delta, \Lambda, A \wedge B}\ \wedge: r$$

- $\vee$ introduction:

$$\frac{A, \Gamma \vdash \Delta \qquad B, \Pi \vdash \Lambda}{A \vee B, \Gamma, \Pi \vdash \Delta, \Lambda}\ \vee: l$$

$$\frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, A \vee B}\ \vee: r1 \qquad \frac{\Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \vee B}\ \vee: r2$$

Propositional rules (ctd.)

- $\neg$ introduction:

$$\frac{\Gamma \vdash \Delta, A}{\neg A, \Gamma \vdash \Delta}\ \neg : l \qquad \frac{A, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \neg A}\ \neg : r$$

- $\supset$ introduction:

$$\frac{\Gamma \vdash \Delta, A \qquad B, \Pi \vdash \Lambda}{A \supset B, \Gamma, \Pi \vdash \Delta, \Lambda}\ \supset : l \qquad \frac{A, \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \supset B}\ \supset : r$$

Quantifier rules

- $\forall$ introduction:

$$\frac{A(t), \Gamma \vdash \Delta}{(\forall x)A(x), \Gamma \vdash \Delta}\ \forall: l \qquad \frac{\Gamma \vdash \Delta, A(\alpha)}{\Gamma \vdash \Delta, (\forall x)A(x)}\ \forall: r$$

- $\exists$ introduction:

$$\frac{A(\alpha), \Gamma \vdash \Delta}{(\exists x)A(x), \Gamma \vdash \Delta}\ \exists: l \qquad \frac{\Gamma \vdash \Delta, A(t)}{\Gamma \vdash \Delta, (\exists x)A(x)}\ \exists: r$$

Structural rules

- Weakening rules:

$$\frac{\Gamma \vdash \Delta}{A, \Gamma \vdash \Delta}\, w\colon l \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, A}\, w\colon r$$

- Contraction rules:

$$\frac{A, A, \Gamma \vdash \Delta}{A, \Gamma \vdash \Delta}\, c\colon l \qquad \frac{\Gamma \vdash \Delta, A, A}{\Gamma \vdash \Delta, A}\, c\colon r$$

Cut rule

- The cut rule:

$$\frac{\Gamma \vdash \Delta, A \qquad A, \Pi \vdash \Lambda}{\Gamma, \Pi \vdash \Delta, \Lambda}\, cut$$

- The only rule such that its upper sequents may contain formula occurrences that do not appear in the lower sesuents.
- The only rule that may produce an inconsistency ($\vdash$).
- The upper sequents of a *cut* rule corresponds to the lemmas into the proof.

Resolution Calculus

- Clauses are atomic sequents.
- Resolution rule is a cut rule on clauses, where cut-formulas can be unified with an m.g.u σ.
- Factorization rule is a contraction rule on clauses, where contracted formulas can be unified with an m.g.u σ.
- Resolution deduction is a derivation tree having clauses as nodes and resolution, factorization and weakening rules as edges.
- Resolution refutation is a resolution derivation of $\vdash$.

Methods of Cut-elimination

Gentzen's Method

- Gentzen's method of cut-elimination is reductive, i.e. proof rewriting system is defined which is terminating and its normal form is a cut-free proof.
- Rewriting rules are divided into two parts: grade and rank reduction rules.
- Grade of a cut rule is the number of logical symbols in the cut-formula.
- Rank of a cut rule is the number of occurrences of cut-formulas in the left and right cut-derivation.

The CERES Method

- CERES is a cut-elimination method by resolution.
- Method consists of the following steps:
  1. Skolemization of the proof (if it is not already skolemized).
  2. Computation of the characteristic clause set.
  3. Refutation of the characteristic clause set.
  4. Computation of the Projections and construction of the Atomic Cut Normal Form.

The CERES Method (ctd.)

- if ρ is an axiom of the form $\Gamma_C, \Gamma \vdash \Delta_C, \Delta$, then
$$\mathrm{CL}_\rho(\psi) = \{\Gamma_C \vdash \Delta_C\}\,.$$
- if ρ is an unary rule with immediate predecessor ρ', then
$$\mathrm{CL}_\rho(\psi) = \mathrm{CL}_{\rho'}(\psi).$$
- if ρ is a binary rule with immediate predecessors ρ_1, ρ_2, then either
$$\mathrm{CL}_\rho(\psi) = \mathrm{CL}_{\rho_1}(\psi) \cup \mathrm{CL}_{\rho_2}(\psi)$$
or
$$\mathrm{CL}_\rho(\psi) = \mathrm{CL}_{\rho_1}(\psi) \otimes \mathrm{CL}_{\rho_2}(\psi).$$
- $\mathrm{CL}(\psi) = \mathrm{CL}_{\rho_0}(\psi)$.

An Example (thanks to D. Weller)

$$\varphi_1 = \dfrac{\dfrac{\dfrac{P(u) \vdash P(u) \quad Q(u) \vdash Q(u)}{P(u) \supset Q(u) \vdash (\exists y)(P(u) \supset Q(y))}\ \exists : r, \supset: r, \supset: l}{(\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(u) \supset Q(y))}\ \forall : l}{(\forall x)(P(x) \supset Q(x)) \vdash (\forall x)(\exists y)(P(x) \supset Q(y))}\ \forall : r$$

$$\varphi_2 = \dfrac{\dfrac{\dfrac{P(a) \vdash P(a) \quad Q(v) \vdash Q(v)}{P(a) \supset Q(v) \vdash (\exists y)(P(a) \supset Q(y))}\ \exists : r, \supset: r, \supset: l}{(\exists y)(P(a) \supset Q(y)) \vdash (\exists y)(P(a) \supset Q(y))}\ \exists : l}{(\forall x)(\exists y)(P(x) \supset Q(y)) \vdash (\exists y)(P(a) \supset Q(y))}\ \forall : l$$

$$\varphi = \dfrac{\varphi_1 \qquad \varphi_2}{(\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(a) \supset Q(y))}\ cut$$

An Example (thanks to D. Weller)

$$\varphi_1 = \dfrac{\dfrac{\dfrac{P(u) \vdash P(u) \quad Q(u) \vdash Q(u)}{P(u) \supset Q(u) \vdash (\exists y)(P(u) \supset Q(y))}\ \exists : r, \supset : r, \supset : l}{(\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(u) \supset Q(y))}\ \forall : l}{(\forall x)(P(x) \supset Q(x)) \vdash (\forall x)(\exists y)(P(x) \supset Q(y))}\ \forall : r$$

$$\varphi_2 = \dfrac{\dfrac{\dfrac{P(a) \vdash P(a) \quad Q(v) \vdash Q(v)}{P(a) \supset Q(v) \vdash (\exists y)(P(a) \supset Q(y))}\ \exists : r, \supset : r, \supset : l}{(\exists y)(P(a) \supset Q(y)) \vdash (\exists y)(P(a) \supset Q(y))}\ \exists : l}{(\forall x)(\exists y)(P(x) \supset Q(y)) \vdash (\exists y)(P(a) \supset Q(y))}\ \forall : l$$

$$\varphi = \dfrac{\varphi_1 \qquad \varphi_2}{(\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(a) \supset Q(y))}\ cut$$

An Example (thanks to D. Weller)

$$\varphi_1 = \cfrac{\cfrac{\cfrac{P(u) \vdash P(u) \quad Q(u) \vdash Q(u)}{P(u) \supset Q(u) \vdash (\exists y)(P(u) \supset Q(y))} \; \exists : r, \supset : r, \supset : l}{(\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(u) \supset Q(y))} \; \forall : l}{(\forall x)(P(x) \supset Q(x)) \vdash (\forall x)(\exists y)(P(x) \supset Q(y))} \; \forall : r$$

$$\varphi_2 = \cfrac{\cfrac{\cfrac{P(a) \vdash P(a) \quad Q(v) \vdash Q(v)}{P(a) \supset Q(v) \vdash (\exists y)(P(a) \supset Q(y))} \; \exists : r, \supset : r, \supset : l}{(\exists y)(P(a) \supset Q(y)) \vdash (\exists y)(P(a) \supset Q(y))} \; \exists : l}{(\forall x)(\exists y)(P(x) \supset Q(y)) \vdash (\exists y)(P(a) \supset Q(y))} \; \forall : l$$

$$\varphi = \cfrac{\varphi_1 \qquad \varphi_2}{\vdash} \; cut$$

An Example (thanks to D. Weller)

$$\varphi_1 = \cfrac{\cfrac{\cfrac{P(u) \vdash P(u) \quad Q(u) \vdash Q(u)}{P(u) \supset Q(u) \vdash (\exists y)(P(u) \supset Q(y))}\ \exists : r, \supset : r, \supset : l}{(\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(u) \supset Q(y))}\ \forall : l}{(\forall x)(P(x) \supset Q(x)) \vdash (\forall x)(\exists y)(P(x) \supset Q(y))}\ \forall : r$$

$$\varphi_2 = \cfrac{\cfrac{\cfrac{\vdash P(a) \quad Q(v) \vdash}{P(a) \supset Q(v) \vdash}\ \supset : l}{(\exists y)(P(a) \supset Q(y)) \vdash}\ \exists : l}{(\forall x)(\exists y)(P(x) \supset Q(y)) \vdash}\ \forall : l$$

$$\varphi = \cfrac{\varphi_1 \qquad \varphi_2}{\vdash}\ cut$$

An Example (thanks to D. Weller)

$$\varphi_1 = \dfrac{\dfrac{\dfrac{P(u) \vdash Q(u)}{\vdash (\exists y)(P(u) \supset Q(y))}\ \exists : r, \supset : r}{\vdash (\exists y)(P(u) \supset Q(y))}}{\vdash (\forall x)(\exists y)(P(x) \supset Q(y))}\ \forall : r$$

$$\varphi_2 = \dfrac{\dfrac{\dfrac{\vdash P(a) \quad Q(v) \vdash}{P(a) \supset Q(v) \vdash}\ \supset : l}{(\exists y)(P(a) \supset Q(y)) \vdash}\ \exists : l}{(\forall x)(\exists y)(P(x) \supset Q(y)) \vdash}\ \forall : l$$

$$\varphi = \dfrac{\varphi_1 \qquad \varphi_2}{\vdash}\ cut$$

An Example (thanks to D. Weller)

$$\mathrm{CL}(\varphi) = \{P(u) \vdash Q(u);\ \vdash P(a);\ Q(v) \vdash\}$$

refutation:

$$\dfrac{\dfrac{\vdash P(a) \quad P(u) \vdash Q(u)}{\vdash Q(a)}\ R \qquad Q(v) \vdash}{\vdash}\ R$$

$\sigma = \{u \leftarrow a, v \leftarrow a\}$
ground refutation:

$$\dfrac{\dfrac{\vdash P(a) \quad P(a) \vdash Q(a)}{\vdash Q(a)}\ R \qquad Q(a) \vdash}{\vdash}\ R$$

An Example (thanks to D. Weller)

$$\varphi_1 = \dfrac{\dfrac{\dfrac{P(u) \vdash P(u) \quad Q(u) \vdash Q(u)}{P(u) \supset Q(u) \vdash (\exists y)(P(u) \supset Q(y))}\ \exists : r, \supset : r, \supset : l}{(\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(u) \supset Q(y))}\ \forall : l}{(\forall x)(P(x) \supset Q(x)) \vdash (\forall x)(\exists y)(P(x) \supset Q(y))}\ \forall : r$$

$$\varphi_2 = \dfrac{\dfrac{\dfrac{P(a) \vdash P(a) \quad Q(v) \vdash Q(v)}{P(a) \supset Q(v) \vdash (\exists y)(P(a) \supset Q(y))}\ \exists : r, \supset : r, \supset : l}{(\exists y)(P(a) \supset Q(y)) \vdash (\exists y)(P(a) \supset Q(y))}\ \exists : l}{(\forall x)(\exists y)(P(x) \supset Q(y)) \vdash (\exists y)(P(a) \supset Q(y))}\ \forall : l$$

$$\varphi = \dfrac{\varphi_1 \qquad \varphi_2}{(\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(a) \supset Q(y))}\ cut$$

An Example (thanks to D. Weller)

$$\varphi_1 = \dfrac{\dfrac{\dfrac{P(u) \vdash P(u) \quad Q(u) \vdash Q(u)}{P(u) \supset Q(u), P(u) \vdash Q(u)} \supset : l}{(\forall x)(P(x) \supset Q(x)), P(u) \vdash Q(u)} \forall : l}{(\forall x)(P(x) \supset Q(x)), P(u) \vdash Q(u)}$$

$$\varphi_2 = \dfrac{\dfrac{\dfrac{P(a) \vdash P(a) \quad Q(v) \vdash Q(v)}{P(a) \supset Q(v) \vdash (\exists y)(P(a) \supset Q(y))} \exists : r, \supset : r, \supset : l}{(\exists y)(P(a) \supset Q(y)) \vdash (\exists y)(P(a) \supset Q(y))} \exists : l}{(\forall x)(\exists y)(P(x) \supset Q(y)) \vdash (\exists y)(P(a) \supset Q(y))} \forall : l$$

$$\varphi = \dfrac{\varphi_1 \qquad \varphi_2}{(\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(a) \supset Q(y))} \; cut$$

An Example (thanks to D. Weller)

$$\varphi_1 = \cfrac{\cfrac{\cfrac{P(u) \vdash P(u) \quad Q(u) \vdash Q(u)}{P(u) \supset Q(u), P(u) \vdash Q(u)}\ \supset : l}{(\forall x)(P(x) \supset Q(x)), P(u) \vdash Q(u)}\ \forall : l}{(\forall x)(P(x) \supset Q(x)), P(u) \vdash Q(u)}$$

$$\varphi_2 = \cfrac{\cfrac{\cfrac{P(a) \vdash P(a) \quad Q(v) \vdash Q(v)}{P(a) \supset Q(v) \vdash (\exists y)(P(a) \supset Q(y))}\ \exists : r, \supset : r, \supset : l}{(\exists y)(P(a) \supset Q(y)) \vdash (\exists y)(P(a) \supset Q(y))}\ \exists : l}{(\forall x)(\exists y)(P(x) \supset Q(y)) \vdash (\exists y)(P(a) \supset Q(y))}\ \forall : l$$

$$\varphi = \cfrac{\varphi_1 \qquad \varphi_2}{(\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(a) \supset Q(y))}\ cut$$

An Example (thanks to D. Weller)

$$\dfrac{\dfrac{\dfrac{P(u) \vdash P(u) \quad Q(u) \vdash Q(u)}{P(u) \supset Q(u), P(u) \vdash Q(u)} \supset: l}{(\forall x)(P(x) \supset Q(x)), P(u) \vdash Q(u)} \forall : l}{\varphi_1 = (\forall x)(P(x) \supset Q(x)), P(u) \vdash Q(u)}$$

$$\dfrac{\dfrac{\dfrac{P(a) \vdash P(a)}{\vdash P(a), (\exists y)(P(a) \supset Q(y))} \exists : r, \supset: r, w : r}{\vdash P(a), (\exists y)(P(a) \supset Q(y))}}{\varphi_2 = \vdash P(a), (\exists y)(P(a) \supset Q(y))}$$

$$\varphi = \dfrac{\varphi_1 \qquad \varphi_2}{(\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(a) \supset Q(y))} \; cut$$

An Example (thanks to D. Weller)

$$\varphi_1 = \dfrac{\dfrac{\dfrac{P(u) \vdash P(u) \quad Q(u) \vdash Q(u)}{P(u) \supset Q(u), P(u) \vdash Q(u)} \supset: l}{(\forall x)(P(x) \supset Q(x)), P(u) \vdash Q(u)} \forall : l}{(\forall x)(P(x) \supset Q(x)), P(u) \vdash Q(u)}$$

$$\varphi_2 = \dfrac{\dfrac{\dfrac{P(a) \vdash P(a) \quad Q(v) \vdash Q(v)}{P(a) \supset Q(v) \vdash (\exists y)(P(a) \supset Q(y))} \exists : r, \supset: r, \supset: l}{(\exists y)(P(a) \supset Q(y)) \vdash (\exists y)(P(a) \supset Q(y))} \exists : l}{(\forall x)(\exists y)(P(x) \supset Q(y)) \vdash (\exists y)(P(a) \supset Q(y))} \forall : l$$

$$\varphi = \dfrac{\varphi_1 \qquad \varphi_2}{(\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(a) \supset Q(y))} \; cut$$

An Example (thanks to D. Weller)

$$\cfrac{\cfrac{\cfrac{P(u) \vdash P(u) \quad Q(u) \vdash Q(u)}{P(u) \supset Q(u), P(u) \vdash Q(u)} \supset: l}{(\forall x)(P(x) \supset Q(x)), P(u) \vdash Q(u)} \forall : l}{\varphi_1 = (\forall x)(P(x) \supset Q(x)), P(u) \vdash Q(u)}$$

$$\cfrac{\cfrac{\cfrac{Q(v) \vdash Q(v)}{Q(v) \vdash (\exists y)(P(a) \supset Q(y))} \exists : r, \supset: r, w : l}{Q(v) \vdash (\exists y)(P(a) \supset Q(y))}}{\varphi_2 = Q(v) \vdash (\exists y)(P(a) \supset Q(y))}$$

$$\varphi = \cfrac{\varphi_1 \qquad \varphi_2}{(\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(a) \supset Q(y))} \; cut$$

An Example (thanks to D. Weller)

$\varphi(P(a) \vdash Q(a)) =$

$$\dfrac{\dfrac{\dfrac{P(a) \vdash P(a) \quad Q(a) \vdash Q(a)}{P(a), P(a) \supset Q(a) \vdash Q(a)} \supset: l}{P(a), (\forall x)(P(x) \supset Q(x)) \vdash Q(a)} \forall : l}{P(a), (\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(a) \supset Q(y)), Q(a)} w : r$$

$\varphi(\vdash P(a)) =$

$$\dfrac{\dfrac{\dfrac{\dfrac{P(a) \vdash P(a)}{P(a) \vdash P(a), Q(v)} w : r}{\vdash P(a) \supset Q(v), P(a)} \supset: r}{\vdash (\exists y)(P(a) \supset Q(y)), P(a)} \exists : r}{(\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(a) \supset Q(y)), P(a)} w : l$$

An Example (thanks to D. Weller)

$\varphi(Q(a) \vdash) =$

$$\dfrac{\dfrac{\dfrac{\dfrac{Q(a) \vdash Q(a)}{P(a), Q(a) \vdash Q(a)}\ w:l}{Q(a) \vdash P(a) \supset Q(a)}\ \supset: r}{Q(a) \vdash (\exists y)(P(a) \supset Q(y))}\ \exists : r}{Q(a), (\forall x)(P(x) \supset Q(x)) \vdash (\exists y)(P(a) \supset Q(y))}\ w:l$$

An Example (thanks to D. Weller)

$\varphi(\gamma) =$

$$
\dfrac{\dfrac{\dfrac{\varphi(\vdash P(a))}{B \vdash C, P(a)} \quad \dfrac{\varphi(P(a) \vdash Q(a))}{P(a), B \vdash C, Q(a)}}{B, B \vdash C, C, Q(a)}\ cut \quad \dfrac{\varphi(\vdash Q(a))}{Q(a), B \vdash C}}{\dfrac{B, B, B \vdash C, C, C}{B \vdash C}\ \text{contractions}}\ cut
$$

where $B = (\forall x)(P(x) \supset Q(x))$, $C = (\exists y)(P(a) \supset Q(y))$.

Fast Cut-Elimination

What is fast cut-elimination?

- A function $f : \mathbb{N} \to \mathbb{N}$ is called elementary, iff computing time of f is bounded by an exponential function, i.e.

$$2^{2^{\cdot^{\cdot^{\cdot^{2^n}}}}}\Big\}_m$$

- CERES is fast on the subclass of **LK**-proofs Φ, i.e. cut-elimination is elementary on Φ, iff resolution complexity of the characteristic clause set is bound by an elementary function.

- Idea: identify classes where CERES is fast, i.e. cut-elimination is elementary.

How?

- Decidable subclasses of FOL:
  - Herbrand class: $(Q\vec{x})(L_1 \wedge \ldots \wedge L_m)$.
  - Bernays - Schönfinkel class: $(\exists\vec{x})(\forall\vec{y})M$.
  - Ackermann class: $(\exists\vec{x})(\forall y)(\exists\vec{z})M$.
  - One-variable class: $|Var(F)| \leq 1$.
  - Monadic class: formulas contain only unary predicate symbols.
- Restrict inference rules or syntax of cut-formulas s.t. characteristic clause set falls into one of these classes.

Fast cut-elimination classes

- The following classes of **LK**-proofs are fast:

Class UIE:

- All inferences that go into the end-sequent are unary.

Complexity: exponential.

Fast cut-elimination classes

- The following classes of **LK**-proofs are fast:

Class UILM:

1. Only one monotone cut.
2. All inferences in the left cut-derivation that go into the end-sequent are unary.

Complexity: double exponential.

Fast cut-elimination classes

- The following classes of **LK**-proofs are fast:

Class UIRM:

1. Only one monotone cut.
2. All inferences in the right cut-derivation that go into the end-sequent are unary.

Complexity: double exponential.

Fast cut-elimination classes

- The following classes of **LK**-proofs are fast:

Class AXDC:

- Different axioms are variable disjoint.

Complexity: double exponential.

Fast cut-elimination classes

- The following classes of **LK**-proofs are fast:

Class MC:

- All function and predicate symbols appearing in cut-formulas are monadic.

Complexity: double exponential.

Fast cut-elimination classes (ctd.)

- We have shown that the following classes are fast:

  Class G-UILM:

  1. All cuts are monotone.
  2. All inferences in all left cut-derivation that go into the end-sequent are unary.
  3. No binary rule, that goes into the end sequent, connects two cuts.

Fast cut-elimination classes (ctd.)

- We have shown that the following classes are fast:

Class G-UILM:

1. All cuts are monotone.
2. All inferences in all left cut-derivation that go into the end-sequent are unary.
3. No binary rule, that goes into the end sequent, connects two cuts.

$$((\vdash A_1^1 \oplus \ldots \oplus \vdash A_{n_1}^1) \oplus (\otimes_{j_1}(\oplus_{i_1} B_{j_1}^{i_1} \vdash)))$$
$$\oplus \ldots \oplus$$
$$((\vdash A_1^k \oplus \ldots \oplus \vdash A_{n_k}^k) \oplus (\otimes_{j_k}(\oplus_{i_k} B_{j_k}^{i_k} \vdash)))$$

Fast cut-elimination classes (ctd.)

- We have shown that the following classes are fast:

Class G-UILM:

1. All cuts are monotone.
2. All inferences in all left cut-derivation that go into the end-sequent are unary.
3. No binary rule, that goes into the end sequent, connects two cuts.

$$((\vdash A_1^1 \oplus \ldots \oplus \vdash A_{n_1}^1) \oplus (\otimes_{j_1}(\oplus_{i_1} B_{j_1}^{i_1} \vdash)))$$
$$\oplus \ldots \oplus$$
$$((\vdash A_1^k \oplus \ldots \oplus \vdash A_{n_k}^k) \oplus (\otimes_{j_k}(\oplus_{i_k} B_{j_k}^{i_k} \vdash)))$$

Complexity: double exponential.

Fast cut-elimination classes (ctd.)

- We have shown that the following classes are fast:

  Class G-UIRM:

  1. All cuts are monotone.
  2. All inferences in all right cut-derivation that go into the end-sequent are unary.
  3. No binary rule, that goes into the end sequent, connects two cuts.

Fast cut-elimination classes (ctd.)

- We have shown that the following classes are fast:

Class G-UIRM:

1. All cuts are monotone.
2. All inferences in all right cut-derivation that go into the end-sequent are unary.
3. No binary rule, that goes into the end sequent, connects two cuts.

$$((\otimes_{j_1}(\oplus_{i_1} \vdash A^{i_1}_{j_1})) \oplus (B^1_1 \vdash \oplus \ldots \oplus B^1_{n_1} \vdash))$$
$$\oplus \ldots \oplus$$
$$((\otimes_{j_k}(\oplus_{i_k} \vdash A^{i_k}_{j_k})) \oplus (B^k_1 \vdash \oplus \ldots \oplus B^k_{n_k} \vdash))$$

Fast cut-elimination classes (ctd.)

- We have shown that the following classes are fast:

Class G-UIRM:

1. All cuts are monotone.
2. All inferences in all right cut-derivation that go into the end-sequent are unary.
3. No binary rule, that goes into the end sequent, connects two cuts.

$$((\otimes_{j_1}(\oplus_{i_1} \vdash A^{i_1}_{j_1})) \oplus (B^1_1 \vdash \oplus \ldots \oplus B^1_{n_1} \vdash))$$
$$\oplus \ldots \oplus$$
$$((\otimes_{j_k}(\oplus_{i_k} \vdash A^{i_k}_{j_k})) \oplus (B^k_1 \vdash \oplus \ldots \oplus B^k_{n_k} \vdash))$$

Complexity: double exponential.

Fast cut-elimination classes (ctd.)

- We have shown that the following classes are fast:

  Class ONEQ:

  - All cut-formulas have at most one quantifier.

  Complexity: double exponential.

Summary

Conclusion

- Proof transformation, in particular cut-elimination, is one of the key techniques of proof theory.
- Cut-elimination is nonelementary but we use CERES method as a tool to identify classes where it is elementary.
- We proved that G-UILM, G-UIRM and ONEQ are fast classes.

References

Matthias Baaz and Alexander Leitsch.
Cut-elimination and Redundancy-elimination by Resolution.
Journal of Symbolic Computation, 29(2):149–176, 2000.

Matthias Baaz and Alexander Leitsch.
Fast Cut-elimination by CERES.
Proofs, Categories and Computations, College Publications, 2010.

Mikheil Rukhaia.
CERES and Fast Cut-Elimination.
VDM Verlag Dr. Müller Publishing House, Saarbrücken, 2011.

Questions?