Logical Methods in Computer Science
Vol. 9(4:24)2013, pp. 1–25
www.lmcs-online.org

Submitted Jan. 29, 2013
Published Dec. 18, 2013

HERBRAND-CONFLUENCE *

STEFAN HETZL [a] AND LUTZ STRASSBURGER [b]

[a] Institute of Discrete Mathematics and Geometry, Vienna University of Technology, Wiedner Hauptstraße 8-10, 1040 Vienna, Austria
e-mail address: stefan.hetzl@tuwien.ac.at

[b] INRIA Saclay – Île-de-France, 1 rue Honoré d'Estienne d'Orves, Bâtiment Alan Turing, Campus de l'École Polytechnique, 91120 Palaiseau, France
e-mail address: lutz@lix.polytechnique.fr

Abstract. We consider cut-elimination in the sequent calculus for classical first-order logic. It is well known that this system, in its most general form, is neither confluent nor strongly normalizing. In this work we take a coarser (and mathematically more realistic) look at cut-free proofs. We analyze which witnesses they choose for which quantifiers, or in other words: we only consider the Herbrand-disjunction of a cut-free proof. Our main theorem is a confluence result for a natural class of proofs: all (possibly infinitely many) normal forms of the non-erasing reduction lead to the same Herbrand-disjunction.

1. Introduction

The constructive content of proofs has always been a central topic of proof theory and it is also one of the most important influences that logic has on computer science. Classical logic is widely used and presents interesting challenges when it comes to understanding the constructive content of its proofs. These challenges have therefore attracted considerable attention, see, for example, [Par92, DJS97, CH00], [BB96], [Urb00, UB00], [BBS02], [Koh08], or [BL00] for different investigations in this direction.

A well-known, but not yet well-understood, phenomenon is that a single classical proof usually allows several different constructive readings. From the point of view of applications this means that we have a choice among different programs that can be extracted. In [RT12] the authors show that two different extraction methods applied to the same proof produce two programs, one of polynomial and one of exponential average-case complexity. This phenomenon is further exemplified by case studies in [Urb00, BHL$^+$05, BHL$^+$08] as well as the asymptotic results [BH11, Het12b]. The reason for this behavior is that classical "proofs often leave algorithmic detail underspecified" [Avi10].

On the level of cut-elimination in the sequent calculus this phenomenon is reflected by the fact that the standard proof reduction without imposing any strategy is not confluent.

2012 ACM CCS: [**Theory of computation**]: Logic—Proof theory; Formal languages and automata theory—Tree languages.

Key words and phrases: proof theory, first-order logic, tree languages, term rewriting, semantics of proofs.

* This paper is an extended version of [HS12].

LOGICAL METHODS IN COMPUTER SCIENCE
DOI:10.2168/LMCS-9(4:24)2013
© S. Hetzl and L. Straßburger
Creative Commons

In this paper we consider cut-elimination in classical first-order logic and treat the question which cut-free proofs one can obtain (by the strategy-free rewriting system) from a single proof with cuts. As our aim is to compare cut-free proofs we need a notion of equivalence of proofs: clearly the syntactic equality makes more differences than those which are mathematically interesting. Being in a system with quantifiers, a natural and more realistic choice is to consider two cut-free proofs equivalent if they choose the same terms for the same quantifiers, in other words: if they have the same Herbrand-disjunction.

A cut-reduction relation will then be called *Herbrand-confluent* if all its normal forms have the same Herbrand-disjunction. The main result of this paper is that, for a natural class of proofs, the standard reduction without erasing of subproofs is Herbrand-confluent. This result is surprising as this reduction is neither confluent nor strongly normalizing and may produce normal forms of arbitrary size (which—as our result shows—arise only from repetitions of the same instances).

As a central proof technique we use rigid tree languages which have been introduced in [JKV09] with applications in verification (e.g. of cryptographic protocols as in [JKV11]) as their primary purpose. To a proof we will associate a rigid tree grammar whose language is invariant under non-erasing cut-elimination and hence equal to the only obtainable Herbrand-disjunction. This property suggests the new notion of *Herbrand-content* of a proof, which is defined as the language of the grammar of the proof, and which is a strong invariant. A side effect of this proof technique is a combinatorial description of how the structure of a cut-free proof is related to that of a proof with cut. Such descriptions are important theoretical results which underlie applications such as algorithmic cut-introduction as in [HLW12, HLRW13].

This paper is an extended version of [HS12], where we have worked in a setting that was restricted to proofs of formulas of the shape $\exists x_1 \cdots \exists x_n\, A$, for A quantifier-free. In this paper we extend the results obtained in [HS12] to proofs of arbitrary end-sequents. For this, we first carry out the central technical work in a setting of skolemized end-sequents, and then extend these results to the general case via deskolemization. This proof strategy is analogous to the proof of the second ε-Theorem from the first ε-Theorem in [HB39].

More precisely, this paper is structured as follows: in Section 2 we briefly review the sequent calculus and cut-elimination for classical first-order logic. In Section 3 we describe regular and rigid tree grammars, which we relate to proofs in Section 4. In Section 5 we prove the main invariance lemma in the skolemized setting, and in Section 6 we establish the necessary techniques and results for lifting the invariance lemma to the general case. This lifting is carried out in Section 7 followed by a discussion of several corollaries such as Herbrand-confluence.

2. Sequent Calculus and Cut-Elimination

For the sake of simplicity, we consider only a one-sided sequent calculus and formulas in negation normal form, but the results can be proved for a two-sided sequent calculus in the same way. Thus, our *formulas* (denoted by A, B, ...) are generated from the literals and the constants $\top$ and $\bot$ via the binary connectives $\wedge$ (*and*) and $\vee$ (*or*) and the quantifiers $\exists$ and $\forall$ in the usual way. The *negation* $\overline{A}$ of a formula A is defined via the usual De Morgan laws. A *sequent* (denoted by Γ, Δ, ...) is a multiset of formulas.

Definition 2.1. A *proof* is a tree of sequents, such that every node forms together with its children an instance of one of the inference rules shown below:

$$\frac{}{A, \overline{A}}\,\mathsf{ax} \qquad \frac{}{\top}\,\top \qquad \frac{\Gamma}{\Gamma, A}\,\mathsf{w} \qquad \frac{\Gamma, A, A}{\Gamma, A}\,\mathsf{c} \qquad \frac{\Gamma, A \qquad \overline{A}, \Delta}{\Gamma, \Delta}\,\mathsf{cut}$$

$$\frac{\Gamma, A, B}{\Gamma, A \vee B}\,\vee \qquad \frac{\Gamma, A \qquad \Delta, B}{\Gamma, \Delta, A \wedge B}\,\wedge \qquad \frac{\Gamma, A[x\backslash\alpha]}{\Gamma, \forall x\, A}\,\forall \qquad \frac{\Gamma, A[x\backslash t]}{\Gamma, \exists x\, A}\,\exists$$

where in the ax-rule A has to be a literal, in the $\forall$-rule the α is called *eigenvariable* and does not appear in $\Gamma, \forall x\, A$, and in the $\exists$-rule the term t does not contain a variable bound in A. We use the notation $[x\backslash\alpha]$ for the substitution that replaces x by the eigenvariable α. Similarly, $[x\backslash t]$ is the substitution that replaces x with t.

The explicitly mentioned formula in a conclusion of an inference rule, like $A \vee B$ for $\vee$ is called *main formula*. Analogously, the explicitly mentioned formulas in the premises of an inference rule, like A and B for $\vee$, are called *auxiliary formulas*. In the context of a concrete derivation we speak about *main* and *auxiliary formula occurrences* of inferences.

Definition 2.2. A proof is called *regular* if different $\forall$-inferences have different eigenvariables.

We use the following convention: We use lowercase Greek letters $\alpha, \beta, \gamma, \delta, \ldots$ for *eigenvariables* in proofs, and $\pi, \psi, \ldots$ for proofs. For a proof π, we write $|\pi|$ for the number of occurrences of inferences in π. Furthermore, we write $\mathrm{EV}(\pi)$ for the set of eigenvariables of $\forall$-inferences of π.

In a sequent calculus proof, each formula occurrence can be traced downwards via its descendants to either a cut formula or the end-sequent. We write $\mathrm{EV}_{\mathrm{c}}(\pi)$ for the set of those eigenvariables in π that are introduced by a $\forall$-inference whose main formula occurrence can be traced downwards to a cut formula, i.e., is not part of the end-sequent of π. The elements of $\mathrm{EV}_{\mathrm{c}}(\pi)$ will also be called *cut-eigenvariables*.

Definition 2.3. A *weak sequent* is a sequent that does not contain any $\forall$-quantifier.

Fact 2.4. If the end-sequent of a proof π is a weak sequent then $\mathrm{EV}(\pi) = \mathrm{EV}_{\mathrm{c}}(\pi)$.

Remark 2.5. Our results do not depend on technical differences in the definition of the calculus (which in classical logic are inessential) such as the choice between multiplicative and additive rules and the differences in the cut-reduction induced by these choices. However, for the sake of precision, we will formally define the cut-reduction used in this paper.

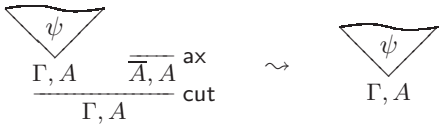
Definition 2.6. Cut-reduction is defined on regular proofs and consists of the proof rewrite steps shown in Figure 1 (as well as all corresponding symmetric variants), where in the contraction reduction step

$$\rho' = [\alpha\backslash\alpha']_{\alpha\in\mathrm{EV}(\psi_2)} \qquad \text{and} \qquad \rho'' = [\alpha\backslash\alpha'']_{\alpha\in\mathrm{EV}(\psi_2)}$$

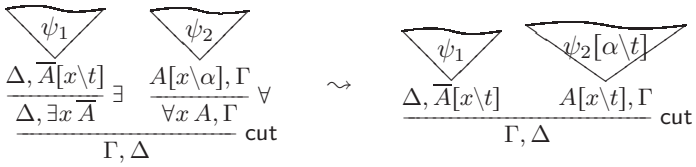
are substitutions replacing each eigenvariable α in ψ_2 by fresh copies, i.e., α' and α'' are fresh for the whole proof. We write $\rightsquigarrow$ for the compatible (w.r.t. the inference rules), reflexive and transitive closure of $\leadsto$.

The above system for cut-reduction consists of purely local, minimal steps and therefore allows the simulation of many other reduction relations. We chose to work in this system in order to obtain invariance results of maximal strength. Among the systems that can

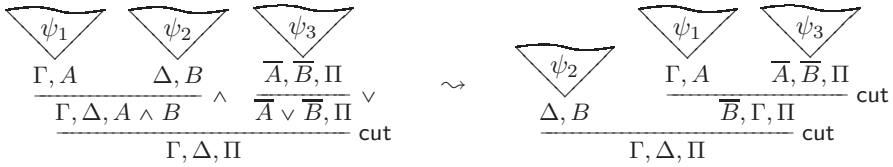
Axiom reduction:



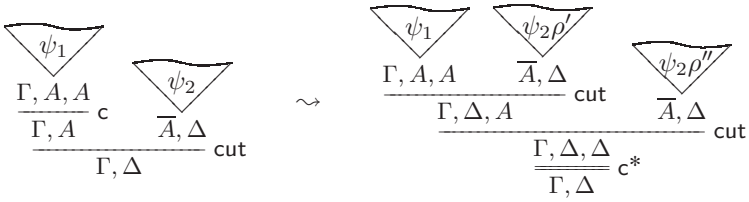
Quantifier reduction:



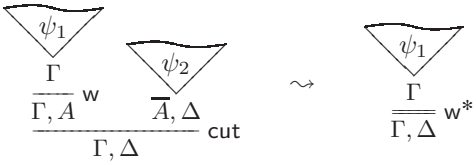
Propositional reduction:



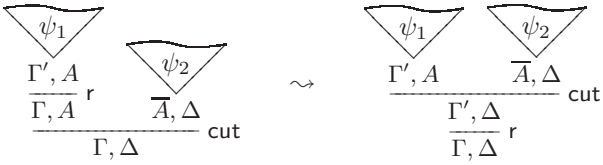
Contraction reduction:



Weakening reduction:



Unary inference permutation:



Binary inference permutation:

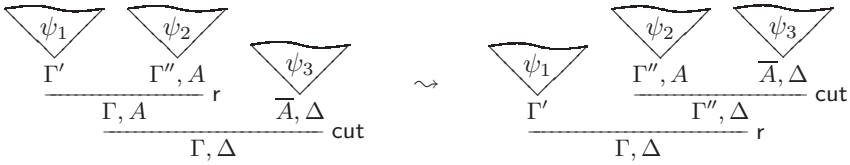


Figure 1: Cut-reduction steps

be simulated literally are for example all color annotations of [DJS97] in the multiplicative version of LK defined there. The real strength of the results in this paper lies however in the general applicability of the used proof techniques: the extraction of a grammar from a proof (that is described in the next sections) is possible in all versions of sequent calculus for classical logic and in principle also in other systems like natural deduction. In particular, our results also apply to inversion-based cut-elimination procedures such as for example that in [Sch77].

3. Regular and Rigid Tree Grammars

Formal language theory constitutes one of the main areas of theoretical computer science. Traditionally, a formal language is defined to be a set of strings but this notion can be generalized in a straightforward way to considering a language to be a set of first-order terms. Such tree languages possess a rich theory and many applications, see e.g. [GS97], [CDG$^+$07]. In this section we introduce notions and results from the theory of tree languages that we will use for our proof-theoretic purposes.

A *ranked alphabet* Σ is a finite set of symbols which have an associated arity (their *rank*). For $f \in \Sigma$, we sometimes use the notation f/n for saying that n is the arity of f. We write $\mathscr{T}_\Sigma$ to denote the set of all finite trees (or terms) over Σ, and we write $\mathscr{T}_\Sigma(X)$ to denote the set of all trees over Σ and a set X of variables (seen as symbols of arity 0). We also use the notion of *position* in a tree, which is a list of natural numbers. We write ε for the empty list (the root position), and we write $p.q$ for the concatenation of lists p and q. We write $p \leqslant q$ if p is a prefix of q and $p < q$ if p is a proper prefix of q. Clearly, $\leqslant$ is a partial order and $<$ is its strict part. We write $\mathrm{Pos}(t)$ to denote the set of all positions in a tree $t \in \mathscr{T}_\Sigma(X)$. Furthermore, for a given tree or term t and position p, we write $t|_p$ to denote the subterm of t that occurs at position p.

Definition 3.1. A *regular tree grammar* is a tuple $G = \langle N, \Sigma, \theta, P\rangle$, where N is a finite set of *non-terminal symbols*, where Σ is a ranked alphabet, such that $N \cap \Sigma = \varnothing$, where θ is the *start symbol* with $\theta \in N$, and where P is a finite set of production rules of the form $\beta \to t$ with $\beta \in N$ and $t \in \mathscr{T}_\Sigma(N)$.

The derivation relation $\to_G$ of a regular tree grammar $G = \langle N, \Sigma, \theta, P\rangle$ is defined as follows. We have $s \to_G r$ if there is a production rule $\beta \to t$ in P and a position $p \in \mathrm{Pos}(s)$, such that $s|_p = \beta$ and r is obtained from s by replacing β at p by t. The *language* of G is then defined as $L(G) = \{t \in \mathscr{T}_\Sigma \mid \theta \to_G^* t\}$, where $\to_G^*$ is the reflexive and transitive closure of $\to_G$. A *derivation* $\mathscr{D}$ of a term $t \in L(G)$ is a sequence $t_0 \to_G t_1 \to_G \ldots \to_G t_n$ with $t_0 = \theta$ and $t_n = t$. Note that a term t might have different derivations in G.

In [JKV09] the class of rigid tree languages has been introduced with applications in verification (e.g. of cryptographic protocols as in [JKV11]) as primary motivation. It will turn out that this class is appropriate for describing cut-elimination in classical first-order logic. In contrast to [JKV09] we do not use automata but grammars—their equivalence is shown in [Het12a].

Definition 3.2. A *rigid tree grammar* is a tuple $\langle N, N_R, \Sigma, \theta, P\rangle$, where $\langle N, \Sigma, \theta, P\rangle$, is a regular tree grammar and $N_R \subseteq N$ is the set of *rigid non-terminals*. We speak of a *totally rigid tree grammar* if $N_R = N$. In this case we will just write $\langle N_R, \Sigma, \theta, P\rangle$.

A derivation $\theta = t_0 \to_G t_1 \to_G \ldots \to_G t_n = t$ of a rigid tree grammar $G = \langle N, N_R, \Sigma, \theta, P\rangle$ is a derivation in the underlying regular tree grammar satisfying the additional *rigidity condition*: If there are $i, j < n$, a non-terminal $\beta \in N_R$, and positions p and q such that $t_i|_p = \beta$ and $t_j|_q = \beta$ then $t|_p = t|_q$. The language $L(G)$ of the rigid tree grammar G is the set of all terms $t \in \mathscr{T}_\Sigma$ which can be derived under the rigidity condition. For a given derivation $\mathscr{D}\colon \theta = t_0 \to_G t_1 \to_G \ldots \to_G t_n = t$ and a non-terminal β we say that $p \in \mathrm{Pos}(t)$ is a *β-position in $\mathscr{D}$* if there is an $i \leqslant n$ with $t_i|_p = \beta$, i.e., either a production rule $\beta \to s$ has been applied at p in $\mathscr{D}$, or β occurs at position p in t. In the context of a given grammar G, we sometimes write $\mathscr{D}\colon \alpha \to_G^* t$ to specify that $\mathscr{D}$ is a derivation starting with α and ending with the term t.

Example 3.3. Let $\Sigma = \{0/0, s/1\}$. A simple pumping argument shows that the language $L = \{f(t,t) \mid t \in \mathscr{T}_\Sigma\}$ is not regular. On the other hand, L is generated by the rigid tree grammar

$$\begin{aligned} G &= \langle\{\theta,\alpha,\beta\},\{\alpha\},\{0/0, s/1, f/2\},\theta,P\rangle \quad \text{where} \\ P &= \{\ \theta \to f(\alpha,\alpha), \\ &\qquad \alpha \to 0 \mid s(\beta), \\ &\qquad \beta \to 0 \mid s(\beta)\ \} \end{aligned}$$

Lemma 3.4. *Let $G = \langle N, N_R, \Sigma, \theta, P\rangle$ be a rigid tree grammar and let $t \in L(G)$. Then there is a derivation $\theta \to_G \ldots \to_G t$ which uses at most one β-production for each $\beta \in N_R$.*

Proof. Given any derivation of t, suppose both $\beta \to s_1$ and $\beta \to s_2$ are used at positions p_1 and p_2 respectively. Then by the rigidity condition $t|_{p_1} = t|_{p_2}$ and we can replace the derivation at p_2 by that at p_1 (or the other way round). This transformation does not violate the rigidity condition because it only copies existing parts of the derivation. □

Lemma 3.5. *Let $G = \langle N_R, \Sigma, \theta, P\rangle$ be a totally rigid tree grammar and $\theta \neq \beta \in N_R$, such that there is exactly one t with $\beta \to t$ in P. If $G' = \langle N_R\backslash\{\beta\}, \Sigma, \theta, (P\backslash\{\beta \to t\})[\beta\backslash t]\rangle$ then $L(G) = L(G')$.*

Proof. If a G-derivation of a term s uses β, it must replace β by t hence s is derivable using the productions of G' as well. The rigidity condition is preserved as the equality constraints of the G'-derivation are a subset of those of the G-derivation. Conversely, given a G'-derivation of a term s we obtain a derivation of s from the productions of G by replacing applications of $\delta \to r[\beta\backslash t]$ by $\delta \to r$ followed by a copy of $\beta \to t$ for each occurrence of β in r. Let $\gamma_1, \ldots, \gamma_n$ be the non-terminals that appear in t. By the rigidity condition for $i \in \{1, \ldots, n\}$ there is a unique term at all γ_i-positions in the derivation. Hence β fulfills the rigidity condition as well, and we have obtained a G-derivation of s. □

Notation 3.6. For a given non-terminal β and a term t, we will write $\beta \in t$ or $t \ni \beta$ for denoting that β occurs in t.

Definition 3.7. Let G be a tree grammar. A *path* of G is a list $\mathscr{P}$ of productions $\alpha_1 \to t_1$, $\ldots$, $\alpha_n \to t_n$ with $n \geqslant 1$ and $\alpha_{i+1} \in t_i$ for all $i \in \{1, \ldots, n-1\}$. The *length* of a path is $|\mathscr{P}| = n$. We will also write $\mathscr{P}\colon \alpha_1 \to t_1 \ni \alpha_2 \to \ldots \ni \alpha_n \to t_n$ to denote a path.

For a given path $\mathscr{P}\colon \alpha_1 \to t_1 \ni \alpha_2 \to \ldots \ni \alpha_n \to t_n$ we say that $\alpha_1, \ldots, \alpha_n$ are *on the path* $\mathscr{P}$ and write $\alpha_i \in \mathscr{P}$ for that. We also write $\mathscr{P}\colon \alpha_1 \dashrightarrow t_n$ and $\mathscr{P}\colon \alpha_1 \dashrightarrow \alpha_n$, if we do not want to explicitly mention the intermediate steps. For a fixed grammar G, we write $\alpha \dashrightarrow \beta$ to denote that there is a path $\mathscr{P}$ in G with $\mathscr{P}\colon \alpha \dashrightarrow \beta$.

For a set P of production rules, we write $\alpha \prec_P \beta$ (or simply $\alpha \prec \beta$, when P is clear from context) if there is a production $\alpha \to t$ in P with $\beta \in t$. We write $\prec^+$ for the transitive closure of $\prec$, and $\prec^*$ for its reflexive, transitive closure. Note that $\alpha \dashrightarrow \beta$ implies $\alpha \prec^+ \beta$, but not the other way around, since β could be a non-terminal with no production $\beta \to s$ in P.

Definition 3.8. A tree grammar $\langle N, \Sigma, \theta, P\rangle$ is called *cyclic* if $\alpha \prec_P^+ \alpha$ for some $\alpha \in N$, and *acyclic* otherwise.

Lemma 3.9. *If G is totally rigid and acyclic, then we have that up to renaming of the non-terminals $G = \langle\{\alpha_1, \ldots, \alpha_n\}, \Sigma, \alpha_1, P\rangle$ with $L(G) = \{\alpha_1[\alpha_1\backslash t_1]\cdots[\alpha_n\backslash t_n] \mid \alpha_i \to t_i \in P\}$.*

Proof. Acyclicity permits a renaming of non-terminals, such that $\alpha_i \prec_P^+ \alpha_j$ implies $i < j$. Then $L(G) \supseteq \{\alpha_1[\alpha_1\backslash t_1]\cdots[\alpha_n\backslash t_n] \mid \alpha_i \to t_i \in P\}$ is obvious. For the left-to-right inclusion, let $\mathscr{D}\colon \alpha_1 = s_1 \to_G \ldots \to_G s_n = s \in \mathscr{T}_\Sigma$ be a derivation in G. By Lemma 3.4 we can assume that for each j at most one production whose left-hand side is α_j is applied, say $\alpha_j \to t_j$. By acyclicity we can rearrange the derivation so that $\alpha_j \to t_j$ is only applied after $\alpha_i \to t_i$ for all $i < j$. For those α_j which do not appear in the derivation we can insert any substitution without changing the final term so we obtain $s = \alpha_1[\alpha_1\backslash t_1]\cdots[\alpha_n\backslash t_n]$. □

This lemma entails that $|L(G)| \leqslant \prod_{i=1}^n |\{t \mid \alpha_i \to t \in P\}|$, in particular we are dealing with a finite language. The central questions in this context are (in contrast to the standard setting in formal language theory) not concerned with *representability* but with the *size of a representation*.

4. Proofs and Grammars

In this section we will relate sequent calculus proofs to rigid tree grammars. A central tool for establishing this relation is Herbrand's theorem [Her30, Bus95]. In its simplest form it states that $\exists x\, A$, for A quantifier-free, is valid iff there are terms $t_1, \ldots, t_n$ such that $\bigvee_{i=1}^n A[x\backslash t_i]$ is a tautology. Such tautological disjunctions of instances are hence called *Herbrand-disjunctions*. Such a disjunction, or equivalently: the set of terms, can be considered a compact representation of a cut-free proof. The relation to tree grammars is based on the observation that a (finite) set of terms is just a (finite) tree language. While the Herbrand-disjunction of a cut-free proof will be considered a tree language, a proof with cut will give a rise to a grammar and its cut-elimination will be described by the computation of the language of its grammar.

There are different options for extending Herbrand's theorem to non-prenex formulas, e.g. the Herbrand proofs of [Bus95] or the expansion trees of [Mil87]. For our purposes it will be most useful to follow the approach of [BL94].

Definition 4.1. Let π be a proof and let O be a formula occurrence in π. Then we define the *Herbrand-set* $\mathrm{H}(O)$ of O inductively as follows:

- If O is the occurrence of a formula A in an axiom, then $\mathrm{H}(O) = \{A\}$.
- If O is in the conclusion sequent of an inference rule without being its main occurrence, then O has exactly one ancestor O' in one of the premises, and we let $\mathrm{H}(O) = \mathrm{H}(O')$.
- If O is the main occurrence in the conclusion of a $\circ$-rule with $\circ \in \{\wedge, \vee\}$ and with auxiliary occurrences O_1 and O_2, then $\mathrm{H}(O) = \{A \circ B \mid A \in \mathrm{H}(O_1), B \in \mathrm{H}(O_2)\}$.
- If O is the main occurrence in the conclusion of a $\forall$- or $\exists$-rule with auxiliary occurrence O_1 in the premise, then $\mathrm{H}(O) = \mathrm{H}(O_1)$.

- If O is the main occurrence in the conclusion of a w-rule, then $\mathrm{H}(O) = \{\bot\}$.
- If O is the main occurrence in the conclusion of a c-rule with auxiliary occurrences O_1 and O_2 in the premise, then $\mathrm{H}(O) = \mathrm{H}(O_1) \cup \mathrm{H}(O_2)$.

Finally, we define

$$\mathrm{H}(\pi) = \bigcup_{P \in \Gamma} \mathrm{H}(P)$$

where Γ is the end-sequent of π and P ranges over all formula occurrences in Γ.

Besides to the Herbrand-set of a formula occurrence, we also need the set of terms associated with an occurrence of an $\exists$-formula.

Definition 4.2. Let Q be an occurrence of a formula $\exists x\, A$ in a proof. We define the set $\mathrm{tm}(Q)$ of *terms associated with* Q as follows: if Q is introduced as the main formula of a weakening, then $\mathrm{tm}(Q) = \varnothing$. If Q is introduced by an $\exists$-rule

$$\frac{\Gamma, A[x\backslash t]}{\Gamma, \exists x\, A}\,\exists$$

then $\mathrm{tm}(Q) = \{t\}$. If Q is the main formula in the conclusion of a contraction, and Q_1 and Q_2 are the two auxiliary occurrences of the same formula in the premise, then $\mathrm{tm}(Q) = \mathrm{tm}(Q_1) \cup \mathrm{tm}(Q_2)$. In all other cases, an inference with the occurrence Q in the conclusion has a corresponding occurrence Q' of the same formula in one of its premises, and we let $\mathrm{tm}(Q) = \mathrm{tm}(Q')$.

In the following, we will restrict our attention to a certain class of proofs, that we call *simple proofs* below.

Definition 4.3. A proof π is called *simple* if

- it is regular (i.e., different $\forall$-inferences have different eigenvariables),
- every cut in π is of one of the following forms:

$$\frac{\Gamma, B \qquad \overline{B}, \Delta}{\Gamma, \Delta}\,\mathsf{cut} \qquad \text{or} \qquad \frac{\Gamma, \exists x\, B \qquad \dfrac{\overline{B}[x\backslash\alpha], \Delta}{\forall x\, \overline{B}, \Delta}\,\forall}{\Gamma, \Delta}\,\mathsf{cut} \tag{4.1}$$

where B is quantifier-free.

Let us make some remarks on this definition. First, we require regularity which is a necessary assumption in the context of cut-elimination. But since every proof can be trivially transformed into a regular one, this is no real restriction. Second, the requirement of the $\forall$-rule being applied directly above the cut is natural as the rule is invertible. Moreover, any proof which does not fulfill this requirement can be pruned to obtain one that does, by simply permuting $\forall$-inferences down and identifying their eigenvariables when needed. Thus, the only significant restriction is that of disallowing quantifier alternations in the cut formulas. This corresponds to allowing only Σ_1 (or Π_1) formulas in cuts.

We conjecture that our central result can be extended to Σ_n-cuts. However, this will require the development of an adequate class of grammars first (see also Section 8).

Observation 4.4. Simple proofs have the technically convenient property of exhibiting a 1-1 relationship between cut-eigenvariables and cuts. For an eigenvariable $\alpha \in \mathrm{EV}_{\mathrm{c}}(\pi)$ we will therefore write cut_α for the corresponding cut and $\forall_\alpha$ for the inference introducing α (when read from bottom to top).

Definition 4.5. Let π be a simple proof, let $\alpha \in \mathrm{EV}_{\mathrm{c}}(\pi)$, and let Q be the occurrence of the existentially quantified cut-formula in the premise of cut_α. Then we write $\mathrm{B}(\alpha)$ for the set of substitutions $\{\ [\alpha\backslash t] \mid t \in \mathrm{tm}(Q)\ \}$ and we define

$$\mathrm{B}(\pi) = \bigcup_{\alpha \in \mathrm{EV}_{\mathrm{c}}(\pi)} \mathrm{B}(\alpha) \quad .$$

Structures similar to the above $\mathrm{B}(\pi)$ have been investigated also in [Hei10] and [McK13] where they form the basis of proof net like formalisms using local reductions for quantifiers in classical first-order logic. Our aim in this work is however quite different: we use these structures for a global analysis of the sequent calculus.

Definition 4.6. Let π be a simple proof. Then the *grammar of* π is the totally rigid grammar $\mathrm{G}(\pi) = \langle N_R, \Sigma, \theta, P \rangle$ with

$$\begin{aligned} N_R &= \mathrm{EV}_{\mathrm{c}}(\pi) \cup \{\theta\} \\ \Sigma &= \Sigma(\pi) \cup \{\wedge, \vee, \top, \bot\} \\ P &= \{\theta \to A \mid A \in \mathrm{H}(\pi)\} \cup \{\alpha \to t \mid [\alpha\backslash t] \in \mathrm{B}(\pi)\} \end{aligned}$$

where $\Sigma(\pi)$ is the signature of π,[1] the rank of $\wedge$ and $\vee$ is 2, the rank of $\top$ and $\bot$ is 0, and θ does not occur in π.

Lemma 4.7. *If π is a simple proof, then $\mathrm{G}(\pi)$ is acyclic.*

Proof. By induction on the number of cuts in π. The grammar of a cut-free proof is trivially acyclic. For the induction step, let r be the lowest binary inference with subproofs π_1 and π_2 such that either (i) r is a cut or (ii) r is not a cut but both π_1 and π_2 contain at least one cut. Let P, P_1, and P_2 be the set of productions induced by the cuts in π, π_1, π_2, respectively. In case (ii), $\prec_P = \prec_{P_1} \cup \prec_{P_2}$, which is acyclic by induction hypothesis (since $\mathrm{EV}_{\mathrm{c}}(\pi_1) \cap \mathrm{EV}_{\mathrm{c}}(\pi_2) = \varnothing$). In case (i), let P_{r} be the productions induced by the cut r, then $\prec_P = \prec_{P_1} \cup \prec_{P_2} \cup \prec_{P_{\mathsf{r}}}$. By induction hypothesis, $\prec_{P_1}$ and $\prec_{P_2}$ are acyclic and as the cut-formula in r contains at most one quantifier, also $\prec_{P_{\mathsf{r}}}$ is acyclic. Therefore, a cycle in $\prec_P^+$ must be of the form $\alpha_1 \prec_{P_1}^* \beta_1 \prec_{P_{\mathsf{r}}} \alpha_2 \prec_{P_2}^+ \beta_2 \prec_{P_{\mathsf{r}}} \alpha_1$ where $\alpha_1, \beta_1 \in \mathrm{EV}_{\mathrm{c}}(\pi_1)$ and $\alpha_2, \beta_2 \in \mathrm{EV}_{\mathrm{c}}(\pi_2)$. However, r contains only one quantifier and depending on its polarity all productions in P_{r} lead from π_1 to π_2 or from π_2 to π_1 but not both, so $\prec_P$ is acyclic. □

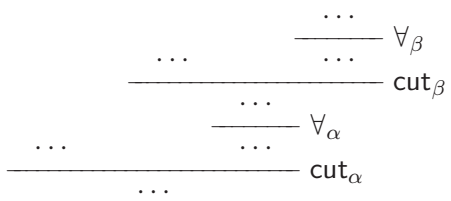
5. Grammars and Cut-Elimination

In this section we will show that the language of the grammar of a proof defined in the previous section is an invariant under cut elimination. Before formulating this invariance result precisely we have to consider the following three aspects of the situation:

First, note that all the reductions shown in Figure 1 preserve simplicity, except the following:

[1] We consider the eigenvariables in $\mathrm{EV}(\pi)\backslash\mathrm{EV}_{\mathrm{c}}(\pi)$ to be part of $\Sigma(\pi)$.

where cut_α is permuted down under cut_β (using the bottommost reduction in Fig. 1) and the cut formula of cut_β has its ancestor on the right side of cut_α. So in the following, when we speak about a *reduction sequence of simple proofs* we require that the above reduction is immediately followed by permuting $\forall_\alpha$ down as well, in order to arrive at



which is again simple.

Secondly, observe that there is no mechanism for deletion in the grammar, but there is one in cut-elimination: the reduction of weakening which erases a sub-proof (see Fig. 1). It is hence natural and will turn out to be useful to also consider the reduction relation without this step.

Definition 5.1. We define the *non-erasing cut-reduction* $\overset{ne}{\rightsquigarrow}$ as $\rightsquigarrow$ without the reduction rule for weakening.

Note that a $\overset{ne}{\rightsquigarrow}$-normal form π is an analytic proof too as $\mathrm{H}(\pi)$ is also a Herbrand-disjunction, i.e. a tautological collection of instances. In contrast to a $\rightsquigarrow$-normal form (which might contain implicit redundancy) a $\overset{ne}{\rightsquigarrow}$-normal form might also contain explicit redundancy in the form of cuts whose cut-formulas are introduced by weakening on one or on both sides. Non-erasing reduction is also of interest in the context of the λ-calculus where it is often considered in the form of the λI-calculus and gives rise to the conservation theorem (see Theorem 13.4.12 in [Bar84]). Our situation here is however quite different: neither $\rightsquigarrow$ nor $\overset{ne}{\rightsquigarrow}$ is confluent and neither of them is strongly normalizing.

Thirdly, in contrast to the case treated in [HS12] in our more general setting it may happen that the reduction of a weakening deletes sub-formulas of formula instances from the proof. In order to treat this situation adequately, we need to define a generalization of the $\subseteq$-relation between sets of formulas. For this reason, we use the symbol $\bot$ for representing subformulas introduced by weakening, a technique also employed in [BHW12, Wel11] for the purpose of a tighter complexity-analysis.

Definition 5.2. The relation $\leqslant$ is defined inductively on quantifier-free formulas as follows:

- for all formulas A we have $\bot \leqslant A$ and $A \leqslant A$, and
- whenever $A' \leqslant A$ and $B' \leqslant B$ then also $A' \wedge B' \leqslant A \wedge B$ and $A' \vee B' \leqslant A \vee B$

Let $\mathscr{A}$ and $\mathscr{B}$ be sets of quantifier-free formulas. Then we define

$$\mathscr{A} \leqslant \mathscr{B} \qquad \text{iff} \qquad \text{for all } A \in \mathscr{A} \text{ there is a } B \in \mathscr{B} \text{ with } A \leqslant B \quad .$$

Fact 5.3. The relation $\leqslant$ is transitive on formula sets.

We are now in a position to precisely state our main invariance lemma which connects grammars with cut-elimination for weak sequents.

Lemma 5.4. *If $\pi \rightsquigarrow \pi'$ is a reduction sequence of simple proofs of a weak sequent, then $L(\mathrm{G}(\pi)) \geqslant L(\mathrm{G}(\pi'))$. If $\pi \overset{ne}{\rightsquigarrow} \pi'$ is a reduction sequence of simple proofs of a weak sequent, then $L(\mathrm{G}(\pi)) = L(\mathrm{G}(\pi'))$.*

The rest of this section is devoted to proving this result. The proof strategy is to carry out an induction on the length of the reduction sequence $\pi \rightsquigarrow \pi'$ (or $\pi \overset{ne}{\rightsquigarrow} \pi'$ respectively) and to make a case distinction on the type of reduction step. The most difficult step will turn out to be the reduction of contraction which duplicates a sub-proof.

Lemma 5.5. *Let π be a simple proof, and let π' be obtained from π by the single application of an axiom reduction, or a propositional reduction, or a unary or binary inference permutation (see Figure 1). Then $L(\mathrm{G}(\pi')) = L(\mathrm{G}(\pi))$.*

Proof. None of these reductions is changing the grammar of the proof, i.e., $\mathrm{G}(\pi') = \mathrm{G}(\pi)$ and therefore also $L(\mathrm{G}(\pi')) = L(\mathrm{G}(\pi))$. □

Lemma 5.6. *Let π be a simple proof, and let π' be obtained from π by the single application of a quantifier reduction (see Figure 1). Then $L(\mathrm{G}(\pi')) = L(\mathrm{G}(\pi))$.*

Proof. Let α be the eigenvariable of the $\forall$-inference and t be the term of the $\exists$-rule directly above the cut that is reduced. Then $\mathrm{G}(\pi')$ can be obtained from $\mathrm{G}(\pi)$ by removing the production rule $\alpha \to t$ and by applying the substitution $[\alpha\backslash t]$ to the right-hand side of all remaining production rules. Thus, $L(\mathrm{G}(\pi')) = L(\mathrm{G}(\pi))$ follows immediately from Lemma 3.5. □

Lemma 5.7. *Let π be a simple proof, and let π' be obtained from π by the single application of a weakening reduction (see Figure 1). Then $L(\mathrm{G}(\pi')) \leqslant L(\mathrm{G}(\pi))$.*

Proof. The grammar $\mathrm{G}(\pi')$ is obtained from $\mathrm{G}(\pi)$ via two modifications. First, all productions coming from cuts or $\exists$-inferences in ψ_2 are deleted, and second, the formulas in Δ which are ancestors of the end-sequent are replaced by $\bot$ in $\mathrm{H}(\pi')$. Now let $A \in L(\mathrm{G}(\pi'))$. Then the derivation of A in $\mathrm{G}(\pi')$ is also a derivation in $\mathrm{G}(\pi)$, with the difference that some $\bot$-subformulas are replaced by other formulas, yielding a formula $B \in L(\mathrm{G}(\pi))$ with $B \geqslant A$. Hence $L(\mathrm{G}(\pi')) \leqslant L(\mathrm{G}(\pi))$. □

It remains to analyze the case of contraction. Surprisingly, also in this case the language of the grammar of a proof remains unchanged. However, the proof of this result is quite technical and requires additional auxiliary results about the relationship between proofs and grammars. Furthermore, this is the case which needs the additional condition that the end-sequent of our proof is weak, i.e., does not contain $\forall$-quantifiers.

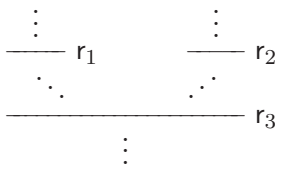
For simplifying the presentation, we assume in the following (without loss of generality) that the $\forall$-side is on the right of a cut and the $\exists$-side on the left. Then, a production $\beta \to t$ in $\mathrm{G}(\pi)$ corresponds to three inferences in π: a cut, an instance of the $\forall$-rule, and an instance of the $\exists$-rule, that we denote by cut_β, $\forall_\beta$, and $\exists_t$, respectively, and that are, in general, arranged in π as shown below.

$$\dfrac{\dfrac{\dfrac{\Gamma', A[x\backslash t]}{\Gamma', \exists x\, A}\,\exists_t}{\vdots}\qquad \dfrac{\dfrac{\overline{A}[x\backslash\beta], \Delta'}{\forall x\, \overline{A}, \Delta'}\,\forall_\beta}{\vdots}}{\dfrac{\Gamma, \exists x\, A \qquad\qquad \forall x\, \overline{A}, \Delta}{\Gamma, \Delta}\,\mathsf{cut}_\beta} \tag{5.1}$$

The additional condition that $\forall_\beta$ is directly above cut_β, as indicated in (4.1) is needed because in the following we make extensive use of Observation 4.4: there is a one-to-one correspondence between the cuts and the eigenvariables in $\mathrm{EV}_{\mathrm{c}}(\pi)$, and thus, the notation cut_β makes sense.

Definition 5.8. We say that the instances cut_β, $\forall_\beta$, and $\exists_t$ *are on a path* $\mathscr{P}$ *in* $\mathrm{G}(\pi)$ if the production $\beta \to t$ is in $\mathscr{P}$.

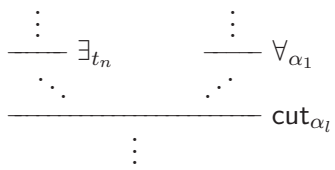
Definition 5.9. Let π be a proof containing the configuration



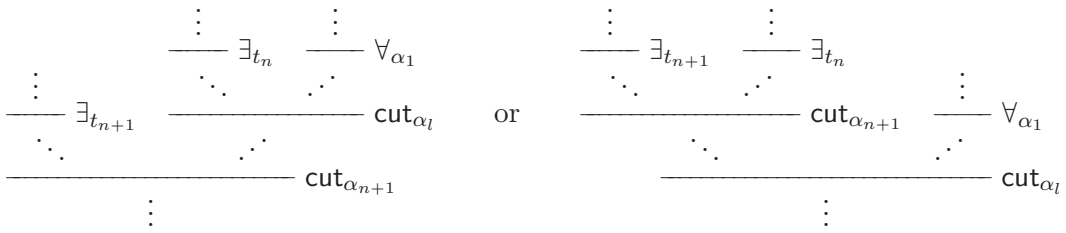
where r_1, r_2, and r_3 are arbitrary rule instances, and r_3 is a branching rule, and r_1 and r_2 might or might not be branching. Then we say that r_1 is *on the left above* r_3, denoted by $\mathsf{r}_1 \leftarrow\!\!\lceil\ \mathsf{r}_3$, and r_2 is *on the right above* r_3, denoted by $\mathsf{r}_3 \rightarrow\!\!\rceil\ \mathsf{r}_2$, and r_1 and r_2 are *in parallel*, denoted by $\mathsf{r}_1 \leftarrow\!\!\lceil\rceil\!\!\rightarrow \mathsf{r}_2$.

Lemma 5.10. *Let π be a simple proof and $\mathscr{P}\colon \alpha_1 \to t_1 \ni \alpha_2 \ldots \to t_n$ be a path in $\mathrm{G}(\pi)$. Then there is a $k \in \{1,\ldots,n\}$ such that cut_{α_k} is lowermost among all inferences on $\mathscr{P}$. Furthermore, $\forall_{\alpha_1}$ is on the right above cut_{α_k} and $\exists_{t_n}$ is on the left above cut_{α_k}.*

Proof. We proceed by induction on n. If $n = 1$, then $n = k = 1$. For the induction step consider a path $\alpha_1 \to t_1 \ni \ldots \ni \alpha_n \to t_n \ni \alpha_{n+1} \to t_{n+1}$. By induction hypothesis, there is some $l \in \{1,\ldots,n\}$ such that we have this configuration



As $\alpha_{n+1} \in t_n$ we know that $\exists_{t_n}$ must be on the right above $\mathsf{cut}_{\alpha_{n+1}}$. Hence, we are in one of the following two situations



In the first case we let $k = n + 1$ and in the second we let $k = l$. In both cases cut_{α_k} has the desired properties. □

Lemma 5.11. *Let π be a simple proof, let $\mathrm{G}(\pi) = \langle N_R, \Sigma, \varphi, P\rangle$, and let $\beta, \alpha \in \mathrm{EV}_\mathrm{c}(\pi)$. If $\beta \dashrightarrow \alpha$ then either $\mathsf{cut}_\alpha \leftarrow\!\!\lceil\ \mathsf{cut}_\beta$ or $\mathsf{cut}_\alpha \rightarrow\!\!\rceil\ \mathsf{cut}_\beta$ or $\mathsf{cut}_\alpha \leftarrow\!\!\lceil\rceil\!\!\rightarrow \mathsf{cut}_\beta$.*

Proof. Since $\beta \dashrightarrow \alpha$, we have a path $\beta \to \ldots \ni \alpha \to t$ for some t. By Lemma 5.10 there is a γ, such that $\exists_t \leftarrow\!\!\lceil\ \mathsf{cut}_\gamma$ and $\mathsf{cut}_\gamma \rightarrow\!\!\rceil\ \forall_\beta$, and such that cut_α and cut_β are not below cut_γ. Furthermore, cut_α must be below $\exists_t$, and cut_β below $\forall_\beta$. If $\gamma = \beta$, then $\mathsf{cut}_\alpha \leftarrow\!\!\lceil\ \mathsf{cut}_\beta$. If $\gamma = \alpha$, then $\mathsf{cut}_\alpha \rightarrow\!\!\rceil\ \mathsf{cut}_\beta$. And if $\gamma \neq \beta$ and $\gamma \neq \alpha$, then $\mathsf{cut}_\alpha \leftarrow\!\!\lceil\rceil\!\!\rightarrow \mathsf{cut}_\beta$. □

Lemma 5.12. *Let* $\mathrm{G}(\pi) = \langle N_R, \Sigma, \varphi, P\rangle$ *be the grammar of a simple proof* π*, such that there are two paths*

$$\beta \to t \ni \gamma_0 \to s_0 \ni \gamma_1 \to s_1 \ni \ldots \to s_{n-1} \ni \gamma_n = \alpha \to s_n$$
$$\beta \to t \ni \delta_0 \to r_0 \ni \delta_1 \to r_1 \ni \ldots \to r_{m-1} \ni \delta_m = \alpha \to r_m$$

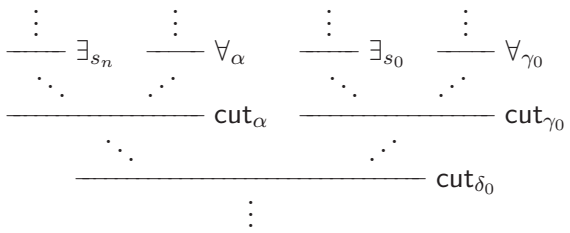
such that γ_0 *and* δ_0 *occur at two different positions in* t*. Then we have one of the following two cases:*

(1) *we have* $\gamma_i = \delta_j$ *for some* $0 \leqslant i < n$ *and* $0 \leqslant j < m$*, or*

(2) *for all* $0 \leqslant i < n$ *and* $0 \leqslant j < m$ *we have* $\mathsf{cut}_\alpha \Rsh \mathsf{cut}_{\gamma_i}$ *and* $\mathsf{cut}_\alpha \Rsh \mathsf{cut}_{\delta_j}$*.*

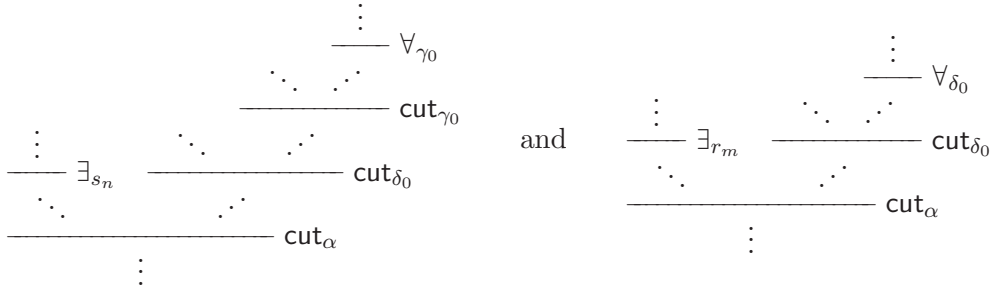
Proof. Note that because of acyclicity of $\mathrm{G}(\pi)$, we have that $\beta \neq \gamma_i$ for all $i \leqslant n$ and $\beta \neq \delta_j$ for all $j \leqslant m$, in particular $\beta \neq \alpha$. Assume, for the moment, that $m, n > 0$; the case of one of them being zero will be treated at the very end of the proof. Then $\gamma_0 \neq \alpha$ and $\delta_0 \neq \alpha$. If $\gamma_0 = \delta_0$, we have case 1. So, assume also $\gamma_0 \neq \delta_0$. As $\beta \to t$ is a production in $\mathrm{G}(\pi)$, the proof π contains a formula which contains both γ_0 and δ_0 hence $\forall_{\gamma_0}$ and $\forall_{\delta_0}$ are not parallel. Since we have $\mathsf{cut}_{\gamma_0} \Rsh \forall_{\gamma_0}$ and $\mathsf{cut}_{\delta_0} \Rsh \forall_{\delta_0}$, we also have that cut_{γ_0} and cut_{δ_0} are not parallel. Without loss of generality, assume that cut_{δ_0} is below cut_{γ_0}. Then $\mathsf{cut}_{\delta_0} \Rsh \mathsf{cut}_{\gamma_0}$ (since $\mathsf{cut}_{\gamma_0} \Lsh \mathsf{cut}_{\delta_0}$ would entail $\forall_{\gamma_0} \Lsh\!\Rsh \forall_{\delta_0}$). Since we have $\delta_0 \dashrightarrow \alpha$, we can apply Lemma 5.11, giving us three possibilities:

- If $\mathsf{cut}_\alpha \Lsh \mathsf{cut}_{\delta_0}$ then we have the situation



  By Lemma 5.10 applied to the path $\gamma_0 \dashrightarrow s_n$ we have that cut_{δ_0} must coincide with cut_{γ_i} for some $0 \leqslant i < n$ (since π is a tree), so $\delta_0 = \gamma_i$ (by Observation 4.4), and we are in case 1.

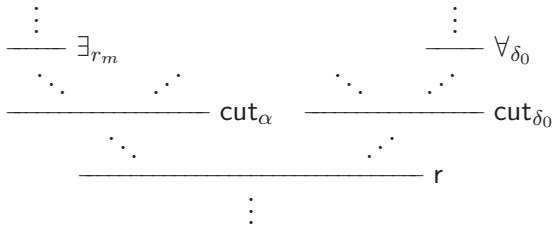
- If $\mathsf{cut}_\alpha \Rsh \mathsf{cut}_{\delta_0}$ then we are in *both* of the following two situations:



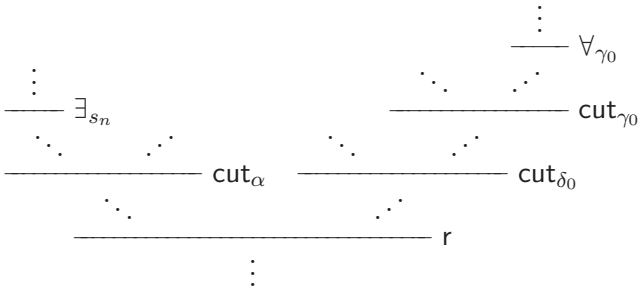
  Thus, by Lemma 5.10 applied to the paths $\gamma_0 \dashrightarrow s_n$ and $\delta_0 \dashrightarrow r_m$ we know that $\mathsf{cut}_\alpha = \mathsf{cut}_{\gamma_k} = \mathsf{cut}_{\delta_l}$ for some $0 \leqslant k \leqslant n$ and $0 \leqslant l \leqslant m$ hence $\gamma_k = \alpha = \delta_l$. Furthermore $k = n$ and $l = m$ by acyclicity of $\mathrm{G}(\pi)$ and assumption $\gamma_n = \alpha = \delta_m$. Now consider any γ_i with $0 \leqslant i < n$. Since $\gamma_i \dashrightarrow \alpha$, we can apply Lemma 5.11 and get either $\mathsf{cut}_\alpha \Lsh \mathsf{cut}_{\gamma_i}$ or $\mathsf{cut}_\alpha \Rsh \mathsf{cut}_{\gamma_i}$ or $\mathsf{cut}_\alpha \Lsh\!\Rsh \mathsf{cut}_{\gamma_i}$. Since by Lemma 5.10 cut_{γ_i} must be above cut_α, we

conclude $\mathsf{cut}_\alpha \Rsh \mathsf{cut}_{\gamma_i}$. With the same reasoning we can conclude that $\mathsf{cut}_\alpha \Rsh \mathsf{cut}_{\delta_j}$ for all $0 \leqslant j < m$. We are therefore in case 2.

- If $\mathsf{cut}_\alpha \leftarrow\!\!\Rsh \mathsf{cut}_{\delta_0}$ then we are in *both* of the following two situations:

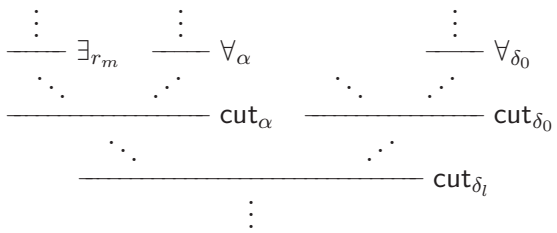


and



By Lemma 5.10 applied to the paths $\gamma_0 \to \ldots \to s_n$ and $\delta_0 \to \ldots \to r_m$, the rule r coincides with cut_{γ_i} and cut_{δ_j} for some $0 < i < n$ and $0 < j < m$, therefore $\gamma_i = \delta_j$ (by Observation 4.4), and we are in case 1.

It remains to treat the case $n = 0$ or $m = 0$. If $m = n = 0$ then we are trivially in case 2 (there is no $0 \leqslant i < n$ or $0 \leqslant j < m$). If $n = 0$ and $m > 0$, we can apply Lemma 5.10 to the path $\delta_0 \to \ldots \to r_m$ and obtain an $l \in \{0, \ldots, m\}$ such that we are in the situation



But by the same argument as at the beginning of the proof, we also have that $\forall_\alpha$ and $\forall_{\delta_0}$ cannot be in parallel (α and δ_0 both appear in t), and therefore either $\mathsf{cut}_{\delta_0} \Rsh \mathsf{cut}_\alpha$ or $\mathsf{cut}_\alpha \Rsh \mathsf{cut}_{\delta_0}$. Since $\delta_0 \dashrightarrow \alpha$, the only possibility is $\mathsf{cut}_\alpha \Rsh \mathsf{cut}_{\delta_0}$, by Lemma 5.11. Thus $\mathsf{cut}_\alpha = \mathsf{cut}_{\delta_l}$, and therefore $l = m$ and we are in case 2. The case $m = 0$ and $n > 0$ is similar. □

We have now finally collected together all necessary tools for describing the reduction step for contraction.

Lemma 5.13. *Let π be a simple proof of a weak sequent such that π contains a subproof ψ, shown on the left below,*

$$\psi \quad = \quad \dfrac{\dfrac{\overset{\psi_1}{\vdots}\ \Gamma, A, A}{\Gamma, A}\,\mathsf{c} \qquad \dfrac{\overset{\psi_2}{\vdots}}{\overline{A}, \Delta}}{\Gamma, \Delta}\,\mathsf{cut} \qquad \rightsquigarrow \qquad \dfrac{\dfrac{\dfrac{\overset{\psi_1}{\vdots}\ \Gamma, A, A \qquad \overset{\psi_2\rho'}{\vdots}\ \overline{A}, \Delta}{\Gamma, \Delta, A}\,\mathsf{cut} \qquad \dfrac{\overset{\psi_2\rho''}{\vdots}}{\overline{A}, \Delta}}{\Gamma, \Delta, \Delta}\,\mathsf{cut}}{\Gamma, \Delta}\,\mathsf{c}^* \qquad = \quad \psi'$$

and let π' be the proof obtained from π from replacing ψ by ψ' shown on the right above, where $\rho' = [\alpha \backslash \alpha']_{\alpha \in \mathrm{EV}(\psi_2)}$ and $\rho'' = [\alpha \backslash \alpha'']_{\alpha \in \mathrm{EV}(\psi_2)}$ are substitutions that replace all eigenvariables in ψ_2 by fresh copies. Then $L(\mathrm{G}(\pi')) = L(\mathrm{G}(\pi))$.

Proof. Let us first show $L(\mathrm{G}(\pi)) \subseteq L(\mathrm{G}(\pi'))$. Write P for the productions of $\mathrm{G}(\pi)$ and P' for those of $\mathrm{G}(\pi')$. Let $F \in L(\mathrm{G}(\pi))$ and $\mathscr{D}$ be its derivation. If the duplicated cut is quantifier-free, then $P' = P\rho' \cup P\rho''$, since the substitutions ρ and ρ' do not affect the eigenvariables outside ψ_2. Hence $\mathscr{D}\rho'$ (as well as $\mathscr{D}\rho''$) is a derivation of F in $\mathrm{G}(\pi')$. If the duplicated cut contains a quantifier, let α be its eigenvariable, let $t_1, \ldots, t_k$ be its terms coming from the left copy of A and $t_{k+1}, \ldots, t_n$ those from the right copy of A and let $Q = \{\alpha \to t_1, \ldots, \alpha \to t_n\} \subseteq P$. We then have

$$P' = (P \backslash Q)\rho' \cup \{\alpha' \to t_1, \ldots, \alpha' \to t_k\} \cup (P \backslash Q)\rho'' \cup \{\alpha'' \to t_{k+1}, \ldots, \alpha'' \to t_n\} \quad .$$

If $\mathscr{D}$ does not contain α, then $\mathscr{D}\rho'$ (as well as $\mathscr{D}\rho''$) is a derivation of F in $\mathrm{G}(\pi')$. If $\mathscr{D}$ does contain α, then by Lemma 3.4 we can assume that it uses only one α-production, say $\alpha \to t_i$. If $1 \leqslant i \leqslant k$, then $\mathscr{D}\rho'$ is a derivation of F in $\mathrm{G}(\pi')$ and if $k < i \leqslant n$, then $\mathscr{D}\rho''$ is a derivation of F in $\mathrm{G}(\pi')$.

Let us now show $L(\mathrm{G}(\pi')) \subseteq L(\mathrm{G}(\pi))$. Let F be a formula in $L(\mathrm{G}(\pi'))$, and let $\mathscr{D}'$ be a derivation of F in $\mathrm{G}(\pi')$. We construct $\mathscr{D} = \mathscr{D}'(\rho')^{-1}(\rho'')^{-1}$ by "undoing" the renaming of the variables in ψ_2. Then $\mathscr{D}$ is a derivation for F, using the production rules of $\mathrm{G}(\pi)$, but possibly violating the rigidity condition.

First, recall that $\mathrm{EV_c}(\pi) = \mathrm{EV}(\pi)$ and observe that only non-terminals $\alpha \in \mathrm{EV}(\psi_2)$ can violate the rigidity condition in $\mathscr{D}$: if $\beta \notin \mathrm{EV}(\psi_2)$ violates the rigidity condition then there are β-positions p_1, p_2 in $\mathscr{D}$ with $F|_{p_1} \neq F|_{p_2}$ and as $\beta\rho'\rho'' = \beta$ the positions p_1, p_2 are also β-positions in $\mathscr{D}'$ and they violate the rigidity condition in $\mathscr{D}'$ which is a contradiction to $\mathscr{D}'$ being a $\mathrm{G}(\pi')$-derivation.

Now define for each $\alpha \in \mathrm{EV}(\psi_2)$ the value $\mathbf{n}(\mathscr{D}, \alpha)$ to be the number of pairs $(p_1, p_2) \in \mathrm{Pos}(F) \times \mathrm{Pos}(F)$ where p_1 and p_2 are α-positions in $\mathscr{D}$ with $p_1 \neq p_2$ and $F|_{p_1} \neq F|_{p_2}$, and define $\mathbf{n}(\mathscr{D}) = \sum_{\alpha \in \mathrm{EV}(\psi_2)} \mathbf{n}(\mathscr{D}, \alpha)$. We proceed by induction on $\mathbf{n}(\mathscr{D})$ to show that $\mathscr{D}$ can be transformed into a derivation which does no longer violate rigidity. If $\mathbf{n}(\mathscr{D}) = 0$ then $\mathscr{D}$ obeys the rigidity condition, and we are done. Otherwise there is at least one $\alpha \in \mathrm{EV}(\psi_2)$ with $\mathbf{n}(\mathscr{D}, \alpha) > 0$. We now pick one such α which is minimal with respect to $\prec^*$ (which exists since $\mathrm{G}(\pi)$ is acyclic). Let p_1 and p_2 be α-positions in $\mathscr{D}$ with $p_1 \neq p_2$ and $F|_{p_1} \neq F|_{p_2}$, let p be the maximal common prefix of p_1 and p_2 and let q be the maximal prefix of p where a production rule has been applied in $\mathscr{D}$. Due to the tree structure of F, the position q is uniquely defined, and q is a β-position for some non-terminal β, and some production rule

$\beta \to t$ has been applied at position q in $\mathscr{D}$, and we have two paths:

$$\beta \to t \ni \gamma_0 \to s_0 \ni \gamma_1 \to s_1 \ni \ldots \to s_{n-1} \ni \gamma_n = \alpha \to s_n$$
$$\beta \to t \ni \delta_0 \to r_0 \ni \delta_1 \to r_1 \ni \ldots \to r_{m-1} \ni \delta_m = \alpha \to r_m$$

where γ_0 and δ_0 occur at two different positions in t. Thus, we can apply Lemma 5.12, giving us the following two cases:

- We have $\gamma_i = \delta_j$ for some $0 \leqslant i < n$ and $0 \leqslant j < m$. Say $\eta = \gamma_i = \delta_j$, and let p_γ and p_δ be the positions of γ_i and δ_j (respectively) in $\mathscr{D}$. Since $\eta <^+ \alpha$ we know that η does not violate the rigidity condition (we chose α to be minimal), and therefore $F|_{p_\gamma} = F|_{p_\delta} = F'$. Let $\mathscr{D}_\gamma \colon \gamma_i \to^*_{\mathrm{G}(\pi)} F'$ and $\mathscr{D}_\delta \colon \delta_j \to^*_{\mathrm{G}(\pi)} F'$ be the two subderivations of $\mathscr{D}$ starting in positions p_γ and p_δ, respectively. Without loss of generality, we can assume that $\mathbf{n}(\mathscr{D}_\gamma) \leqslant \mathbf{n}(\mathscr{D}_\delta)$. Then let $\tilde{\mathscr{D}}$ be the derivation obtained from $\mathscr{D}$ by replacing $\mathscr{D}_\delta$ by $\mathscr{D}_\gamma$. Then $\tilde{\mathscr{D}}$ is still a derivation for F, but $\mathbf{n}(\tilde{\mathscr{D}}) < \mathbf{n}(\mathscr{D})$.
- For all $0 \leqslant i < n$ and $0 \leqslant j < m$ we have $\mathsf{cut}_\alpha \Rsh \mathsf{cut}_{\gamma_i}$ and $\mathsf{cut}_\alpha \Rsh \mathsf{cut}_{\delta_j}$. So all inferences of the path $\gamma_0 \to \ldots \to s_{n-1}$ as well as all inferences of $\delta_0 \to \ldots \to r_{m-1}$ are in ψ_2. Therefore all variables of of these paths are in $\mathrm{EV}(\psi_2)$. As α violates the rigidity in $\mathscr{D}$ one of p_1, p_2 must be a α'-position and the other a α''-position in $\mathscr{D}'$ because $\mathscr{D}'$ does satisfy the rigidity condition. Without loss of generality we can assume that p_1 is the α'-position and p_2 the α''-position. As the paths are contained completely in ψ_2 we have $\gamma_0 \in \mathrm{EV}(\psi_2)\rho'$ and $\delta_0 \in \mathrm{EV}(\psi_2)\rho''$ which is a contradiction as no term can contain both a variable from $\mathrm{EV}(\psi_2)\rho'$ and one from $\mathrm{EV}(\psi_2)\rho''$. □

Proof of Lemma 5.4. By induction on the length of the reduction $\pi \rightsquigarrow \pi'$ or $\pi \overset{ne}{\rightsquigarrow} \pi'$ respectively using one of Lemmas 5.5, 5.6, 5.7 or 5.13 depending on the current reduction step. □

6. Skolemization and Deskolemization

In this section we will describe some results that allow one to extend the above invariance lemma to proofs of arbitrary end-sequents (including $\forall$-quantifiers). Carrying out the above argument directly for arbitrary end-sequents would require dealing with variable-names on the level of the grammar in order to describe the changes of eigenvariables of the $\forall$-quantifiers in the end-sequent. This can be avoided completely by skolemizing proofs to reduce the general case to that of weak sequents and then translating back the results by deskolemization. Skolemization and deskolemization are simple operations on the level of Herbrand-disjunctions or expansion trees [Mil87] and their use in this context suffices for our purposes. In contrast, they have surprising complexity-effects on the level of proofs, see e.g. [BHW12]. The reason why this transfer is possible is that the form of the end-sequent, and in particular the question whether it contains universal quantifiers, does not have an effect on the dynamics of cut-elimination. This observation has been well known for a long time and is apparent already in Gentzen's consistency proof for Peano Arithmetic [Gen38] which is carried out on a (hypothetical) proof of the empty sequent as well as in the proof of the second ε-Theorem from the first ε-Theorem by deskolemization [HB39].

Let us now first define the notion of Herbrand-disjunction precisely. We assume w.l.o.g. that in a formula every variable is bound by at most one quantifier.

Definition 6.1. For a given formula F, we write $\hat{F}$ for the formula obtained from F by removing all quantifiers. Now let $x_1, \ldots, x_n$ be the existentially bound variables in F, and let $y_1, \ldots, y_m$ be the universally bound variables in F. Then any formula of the shape

$$\hat{\hat{F}}[x_1 \backslash t_1, \ldots, x_n \backslash t_n, y_1 \backslash \alpha_1, \ldots, y_m \backslash \alpha_m]$$

where $\hat{\hat{F}}$ is an arbitrary formula with $\hat{\hat{F}} \leqslant \hat{F}$, where $t_1, \ldots, t_n$ are arbitrary terms, and where $\alpha_1, \ldots, \alpha_m$ are fresh variables, is called an *instance of* F. If Γ is a sequent we say that a set $\mathscr{I}$ of formulas is a *set of instances of* Γ if for every $I \in \mathscr{I}$ there is a $F \in \Gamma$, s.t. I is instance of F.

Often we will work in the context of a proof π of a sequent Γ and consider the instances of the formulas in Γ that are induced by π. Then the above fresh variables $\alpha_1, \ldots, \alpha_m$ will be eigenvariables of the proof and their occurrences in terms will be restricted by an acylicity-condition, see below.

Let $\Gamma = F_1, \ldots, F_n$ be a sequent, let $\mathscr{I}$ be a set of instances of Γ, let m_i be the number of quantifiers in F_i, and let l_i be the number of instances of F_i in $\mathscr{I}$. If we impose an arbitrary linear ordering on the instances of F_i in $\mathscr{I}$, then a tuple $\langle i, j, k \rangle$ for $1 \leqslant i \leqslant n$ and $1 \leqslant j \leqslant m_i$ and $1 \leqslant k \leqslant l_i$ uniquely identifies the term which is substituted for the quantifier Qx_j in the k-th instance of the formula F_i. We will write $t_{i,j,k}$ for this term (which could just be an eigenvariable if Qx_j happens to be an $\forall$-quantifier). The k-th instance of F_i can hence be written as $F_{i,k}[x_1 \backslash t_{i,1,k}, \ldots, x_{m_i} \backslash t_{i,m_i,k}]$, where $x_1, \ldots, x_{m_i}$ are the bound variables in F_i, and $F_{i,k}$ is some formula with $F_{i,k} \leqslant \hat{F}_i$. Such a tuple $\langle i, j, k \rangle$ is called *existential position* if x_j is bound existentially in F_i, and *universal position* if x_j is bound universally in F_i.

A position $\langle i_1, j_1, k_1 \rangle$ is said to *dominate* another position $\langle i_2, j_2, k_2 \rangle$, if $i_1 = i_2$, and $k_1 = k_2$, and the quantifier Qx_{j_2} is in the scope of the quantifier Qx_{j_1} in F_i. A set $\mathscr{I}$ of instances induces a relation $\prec$ on its existential positions as: $\langle i_1, j_1, k_1 \rangle \prec \langle i_2, j_2, k_2 \rangle$ if there is a universal position $\langle i_3, j_3, k_3 \rangle$, such that the term t_{i_2,j_2,k_2} contains a variable α with $\alpha = t_{i_3,j_3,k_3}$ and $\langle i_1, j_1, k_1 \rangle$ dominates $\langle i_3, j_3, k_3 \rangle$. Furthermore we define the *dependency relation* $\lessdot$ on the existential positions of $\mathscr{I}$ as transitive closure of $\prec$.

Remark 6.2. A proof π with the property that $\mathrm{H}(\pi) = \mathscr{I}$ is sometimes called a *sequentialization of* $\mathscr{I}$. If $\mathscr{I}$ has positions $\langle i_1, j_1, k_1 \rangle$ and $\langle i_2, j_2, k_2 \rangle$ with $\langle i_1, j_1, k_1 \rangle \prec \langle i_2, j_2, k_2 \rangle$, then in each sequentialization of $\mathscr{I}$ the inference corresponding to $\langle i_1, j_1, k_1 \rangle$ is below that of $\langle i_2, j_2, k_2 \rangle$. In the literature on proof nets, relations like $\prec$ are known as *jumps*.

Definition 6.3. A set $\mathscr{I}$ of instances of Γ is called *Herbrand-disjunction of* Γ if

- the dependency relation $\lessdot$ of $\mathscr{I}$ is acyclic, and
- $\bigvee_{I \in \mathscr{I}} I$ is a tautology.

This notion of Herbrand-disjunction is essentially a flat (as opposed to tree-like) formulation of expansion tree proofs [Mil87]. A similar flat formulation can, for instance, be found in [BL94].

Theorem 6.4. *Γ is valid iff it has a Herbrand-disjunction.*

Proof Sketch. Via translating back and forth with cut-free sequent calculus or alternatively via expansion tree proofs. □

Example 6.5. Let $\Gamma = \exists x\,(\overline{P}(x) \vee \forall y\,P(y))$, let $\mathscr{I} = \{\overline{P}(c) \vee P(\alpha), \overline{P}(\alpha) \vee P(\beta)\}$ and fix the numbering of quantifiers and instances to be from the left to the right. Then there are the two existential positions $\langle 1,1,1\rangle$ with $t_{1,1,1} = c$ and $\langle 1,1,2\rangle$ with $t_{1,1,2} = \alpha$ and two universal positions $\langle 1,2,1\rangle$ with $t_{1,2,1} = \alpha$ and $\langle 1,2,2\rangle$ with $t_{1,2,2} = \beta$. As $\langle 1,1,k\rangle$ dominates $\langle 1,2,k\rangle$, we have $\langle 1,1,1\rangle \prec \langle 1,1,2\rangle$, but not the other way round because $t_{1,1,1} = c$ is variable-free. Therefore $\lessdot$ is acyclic. Furthermore $\mathscr{I}$ is a tautology and hence a Herbrand-disjunction.

Note that for a weak sequent Γ, the induced dependency ordering $\lessdot$ is empty and hence trivially acyclic. The Herbrand-disjunctions of weak sequents are therefore exactly the tautologies of instances.

Definition 6.6. Let $F[\forall y\,G]$ be a formula containing a universal quantifier and let $\exists x_1$, $\ldots$, $\exists x_n$ be the existential quantifiers in whose scope $\forall y$ is. Then define the *Skolemization* of this universal quantifier as

$$\mathrm{sk}_1(F[\forall y\,G]) = F[G[y \backslash g(x_1,\ldots,x_n)]$$

where g is a fresh n-ary function symbol, called a *Skolem function symbol*. The term $g(x_1,\ldots,x_n)$ is called *Skolem-term*. For a formula F define its *Skolemization* $\mathrm{sk}(F)$ to be the iteration of sk_1 until no universal quantifier is left, such that no Skolem function symbol is used for two different universal quantifiers in F. For a sequent $\Gamma = F_1,\ldots,F_n$ define its *Skolemization* $\mathrm{sk}(\Gamma) = \mathrm{sk}(F_1),\ldots,\mathrm{sk}(F_n)$, where no Skolem function symbol is used for two different universal quantifiers in Γ.

Remark 6.7. Sometimes the above operation on formulas is also called Herbrandization. We prefer to use the name Skolemization due to the simple duality between the satisfiability-preserving replacement of existential quantifiers and the validity-preserving replacement of universal quantifiers by new function symbols. There is no danger of confusion as, in the proof-theoretic context of this work, we are clearly dealing with validity only. This use of terminology is due to [HB39], see in particular Section 3.5.a.

The above side condition on the choice of Skolem function symbols results in a 1-1 mapping between universal quantifiers in the sequent we skolemize and the Skolem function symbols. It could be made formally more precise by equipping the sk-operation with such a bijection as second argument. However, for the sake of notational simplicity we refrain from doing so here.

The Skolemization of formulas and sequents can be extended to a Skolemization of proofs. When skolemizing a proof, all universal quantifiers in the end-sequent are removed and their variables are replaced by Skolem-terms. In contrast, the cut-formulas remain unchanged, more precisely:

Definition 6.8. Let π be a proof of a sequent Γ, and let $y_1,\ldots,y_n$ be the variables that are bound by a $\forall$-quantifier in Γ. Furthermore, for each y_i let $\alpha_{i,1},\ldots,\alpha_{i,h_i}$ be the eigenvariables introduced in π by an $\forall$-rule whose main formula is of the shape $\forall y_i\,A$. Then the *Skolemization of the proof* π, denoted by $\mathrm{sk}(\pi)$, is the proof with end-sequent $\mathrm{sk}(\Gamma)$ that is obtained from π by

(1) removing all $\forall$-quantifiers binding one of $y_1,\ldots,y_n$ everywhere, and
(2) replacing each occurrence of y_i (for $i \in \{1,\ldots,n\}$) and $\alpha_{i,j}$ (for $i \in \{1,\ldots,n\}$ and $j \in \{1,\ldots,h_i\}$) by the corresponding Skolem-term. This term is in each case uniquely determined if we proceed from the end-sequent of π upwards to the axioms and demand

that each rule application remains valid, or, in the case of the ∀-rule, becomes void (i.e., premise and conclusion coincide), and
(3) removing the void rule instances.

Note that $\mathrm{sk}(\pi)$ still can contain ∀-quantifiers, namely those coming from a cut.

The Skolemization of a proof π also affects the quantifier-free formulas in π through the replacement of eigenvariables by Skolem terms. In the context of proof Skolemization we hence extend the notation $\mathrm{sk}(\cdot)$ to formulas F from which some (or all) ∀-quantifiers have been removed; then $\mathrm{sk}(F)$ denotes the formula obtained from skolemizing the remaining ∀-quantifiers *and* carrying out the replacement of eigenvariables by Skolem-terms. Skolemization of proofs has the following useful commutation properties.

Lemma 6.9. *If* $\pi \rightsquigarrow \pi'$ *then* $\mathrm{sk}(\pi) \rightsquigarrow \mathrm{sk}(\pi')$. *If* $\pi \overset{ne}{\rightsquigarrow} \pi'$ *then* $\mathrm{sk}(\pi) \overset{ne}{\rightsquigarrow} \mathrm{sk}(\pi')$.

Proof. By induction on the number of reductions in $\pi \rightsquigarrow \pi'$ or $\pi \overset{ne}{\rightsquigarrow} \pi'$, respectively, making a case distinction on the reduction step. The most interesting case is that of the permutation of a ∀-inference over a cut

$$\dfrac{\dfrac{\psi_1}{\dfrac{\Gamma, B[x\backslash\alpha], A}{\Gamma, \forall x\, B, A}\,\forall} \qquad \dfrac{\psi_2}{\overline{A}, \Delta}}{\Gamma, \forall x\, B, \Delta}\,\mathsf{cut} \quad \rightsquigarrow \quad \dfrac{\dfrac{\psi_1}{\Gamma, B[x\backslash\alpha], A} \qquad \dfrac{\psi_2}{\overline{A}, \Delta}}{\dfrac{\Gamma, B[x\backslash\alpha], \Delta}{\Gamma, \forall x\, B, \Delta}\,\forall}\,\mathsf{cut}$$

where the main formula of the ∀-inference is an ancestor of the end-sequent. This reduction step is translated to an identity-step as Skolemization maps both of the above proofs to

$$\dfrac{\dfrac{\psi_1^{\mathrm{s}}}{\mathrm{sk}(\Gamma), \mathrm{sk}(\forall x\, B), \mathrm{sk}(A)} \qquad \dfrac{\psi_2^{\mathrm{s}}}{\mathrm{sk}(\overline{A}), \mathrm{sk}(\Delta)}}{\mathrm{sk}(\Gamma), \mathrm{sk}(\forall x\, B), \mathrm{sk}(\Delta)}\,\mathsf{cut}$$

Each of the other reduction steps translates directly into exactly one reduction step in the skolemized sequence. □

Lemma 6.10. $L(\mathrm{G}(\mathrm{sk}(\pi))) = \mathrm{sk}(L(\mathrm{G}(\pi)))$.

Proof. First note that $\mathrm{EV_c}(\pi) = \mathrm{EV_c}(\mathrm{sk}(\pi))$ hence $\mathrm{G}(\pi)$ and $\mathrm{G}(\mathrm{sk}(\pi))$ have the same non-terminals. Furthermore, to each $\alpha \in \mathrm{EV}(\pi)\backslash\mathrm{EV_c}(\pi)$ corresponds a unique Skolem-term in $\mathrm{sk}(\pi)$, hence to each $F \in \mathrm{H}(\pi)$ and $\sigma \in \mathrm{B}(\pi)$ corresponds a unique $F' \in \mathrm{H}(\mathrm{sk}(\pi))$ and $\sigma' \in \mathrm{B}(\pi)$ and therefore to each production $\alpha \to t$ in $\mathrm{G}(\pi)$ corresponds a unique production $\alpha \to t'$ in $\mathrm{G}(\mathrm{sk}(\pi))$ that is obtained from replacing eigenvariables by their respective Skolem-terms. If $I \in \mathrm{sk}(L(\mathrm{G}(\pi)))$ then by Lemma 3.9 we have $I = \mathrm{sk}(F[\alpha_1\backslash s_1]\cdots[\alpha_n\backslash s_n])$. Now for $\theta \to F, \alpha_1 \to s_1, \ldots, \alpha_n \to s_n$ being the productions in $\mathrm{G}(\pi)$, letting $\theta \to F', \alpha_1 \to s_1', \ldots, \alpha_n \to s_n'$ be the corresponding productions in $\mathrm{G}(\mathrm{sk}(\pi))$ we obtain $F'[\alpha_1\backslash s_1']\cdots[\alpha_n\backslash s_n'] = \mathrm{sk}(F[\alpha_1\backslash s_1]\cdots[\alpha_n\backslash s_n])$. Thus, $\mathrm{sk}(L(\mathrm{G}(\pi))) \subseteq L(\mathrm{G}(\mathrm{sk}(\pi)))$. For the other direction, note that every Skolem-term has at least one corresponding $\alpha \in \mathrm{EV}(\pi)\backslash\mathrm{EV_c}(\pi)$, and as before, this relation translates to productions. So, if $J \in L(\mathrm{G}(\mathrm{sk}(\pi)))$ then by Lemma 3.9 we have $J = G[\alpha_1\backslash t_1]\cdots[\alpha_n\backslash t_n]$ for $\theta \to G, \alpha_1 \to t_1, \ldots, \alpha_n \to t_n$ being the productions in $\mathrm{G}(\mathrm{sk}(\pi))$. By choosing one corresponding set of productions $\theta \to G', \alpha_1 \to t_1', \ldots, \alpha_n \to t_n'$ where Skolem-terms are replaced by the eigenvariables from which they originate we obtain $\mathrm{sk}(G'[\alpha_1\backslash t_1']\cdots[\alpha_n\backslash t_n']) = G[\alpha_1\backslash t_1]\cdots[\alpha_n\backslash t_n]$. □

As we have seen in the above proof, Skolemization can identify instances that differ only in their variable names. The reason for this ability lies in the use of variable names which can be chosen in a redundant way. These superfluous instances can also be removed by an appropriate variable renaming as shown in the following example.

Example 6.11. Let $\Gamma = \exists x \forall y\, (\overline{P}(x,y) \vee \overline{Q}(x,y)), \exists x\, P(c,x) \wedge \exists x\, Q(c,x)$. Then the set of instances obtained from a sequent calculus proof that ends with an $\wedge$-inference is

$$\mathscr{I} = \{\overline{P}(c,\alpha) \vee \overline{Q}(c,\alpha), \overline{P}(c,\beta) \vee \overline{Q}(c,\beta), P(c,\alpha) \wedge Q(c,\beta)\} \quad .$$

Skolemizing would produce the following set of instances

$$\mathrm{sk}(\mathscr{I}) = \{\overline{P}(c,f(c)) \vee \overline{Q}(c,f(c)), P(c,f(c)) \wedge Q(c,f(c))\}$$

by implicitly identifying the two formulas that become equal. A similar effect (but without using Skolemization) can be achieved by directly identifying α and β as in

$$\mathscr{I}[\beta\backslash\alpha] = \{\overline{P}(c,\alpha) \vee \overline{Q}(c,\alpha), P(c,\alpha) \wedge Q(c,\alpha)\} \quad .$$

We now generalize the observations made in the above example. For every Herbrand-disjunction $\mathscr{I}$ there is a substitution ρ, such that $\mathscr{I}\rho$ is a Herbrand-disjunction having the following property: If two universal positions $\langle i,j,k_1\rangle$ and $\langle i,j,k_2\rangle$ have different variables then there is a j', such that the quantifier $\exists x_{j'}$ dominates $\forall x_j$ in F_i and $t_{i,j',k_1} \neq t_{i,j',k_2}$. This follows for example from the formulation of expansion trees in [CHM12a, CHM12b] which use sets of terms for the $\exists$-quantifier and a single variable for the $\forall$-quantifier. A Herbrand-disjunction with this property is α-equivalent to one with *canonical variable names* in the following sense.

Definition 6.12. Let $\mathscr{I}$ be a set of instances. The *canonical name* of the eigenvariable of the universal position $\langle i,j,k\rangle$ is $\alpha_{i,j,t_1,\ldots,t_m}$ where $t_1,\ldots,t_m$ are the terms of the existential positions that dominate $\langle i,j,k\rangle$. The *canonical variable renaming* ρ_{c} of $\mathscr{I}$ is the substitution which replaces all variable names by their canonical names.

Remark 6.13. Note that this relationship is significantly more complex than α-equivalence, as differently named variables are identified according to certain criteria external to variable names. In particular, for some fixed $\mathscr{I}$, there are $\mathscr{I}_n$ of unbounded size such that $\mathscr{I}_n\rho_{\mathrm{c}} = \mathscr{I}$. This can be seen, for example, by continuing Example 6.5: take $\mathscr{I}_n = \{\overline{P}(c) \vee P(\alpha_i), \overline{P}(\alpha_i) \vee P(\beta_i) \mid 1 \leqslant i \leqslant n\}$.

We now turn to *deskolemization*, the inverse operation of Skolemization. In our setting, we only consider deskolemization of sequents and their instances, but not of proofs. Furthermore we always assume that the original sequent with $\forall$-quantifiers is known. Hence the deskolemization of a sequent trivially replaces it by the original sequent. More interesting is the deskolemization of instances which will consist of replacing Skolem-terms by (canonically named) variables.

Definition 6.14. Let $\Gamma = F_1,\ldots,F_n$ be a sequent with Skolem function symbol $f_{i,j}$ for the universal quantifier $\forall x_j$ in F_i. Let $\mathscr{I}$ be a set of instances of Γ and define its *deskolemization* $\mathrm{sk}^{-1}(\mathscr{I})$ by repeating the replacement

$$f_{i,j}(t_1,\ldots,t_m) \mapsto \alpha_{i,j,t_1,\ldots,t_m}$$

on maximal Skolem-terms (w.r.t. the subterm ordering).

In the deskolemization of a Herbrand-disjunction, the acyclicity of the dependency relation is obtained from the acyclicity of the subterm ordering on the Skolem-terms. Conversely, during Skolemization, the Skolem-terms are well-defined due to the acyclicity of the dependency relation (see e.g. [Mil87, Wel11, BHW12] for more details). We hence obtain the following properties:

Lemma 6.15. *Let Γ be a sequent and Γ' be a weak sequent with $\Gamma' = \mathrm{sk}(\Gamma)$.*

(1) *If $\mathscr{I}$ is a Herbrand-disjunction of Γ, then $\mathrm{sk}(\mathscr{I})$ is a Herbrand-disjunction of $\mathrm{sk}(\Gamma)$.*
(2) *If $\mathscr{I}'$ is a Herbrand-disjunction of Γ', then $\mathrm{sk}^{-1}(\mathscr{I}')$ is a Herbrand-disjunction of $\mathrm{sk}^{-1}(\Gamma')$.*
(3) *If $\mathscr{I}$ is a Herbrand-disjunction of Γ, then $\mathrm{sk}^{-1}(\mathrm{sk}(\mathscr{I})) = \mathscr{I}\rho_{\mathrm{c}}$.*

7. Herbrand-Content

Definition 7.1. For a simple proof π, we define its *Herbrand-content* as $[\![\pi]\!] = L(\mathrm{G}(\pi))\rho_{\mathrm{c}}$.

Note that for a cut-free proof π we have $[\![\pi]\!] = \mathrm{H}(\pi)\rho_{\mathrm{c}}$, i.e. the Herbrand-content is nothing other than the Herbrand-disjunction of the proof after variable normalization. Also note that for a proof π of a weak sequent we have $[\![\pi]\!] = L(\mathrm{G}(\pi))$, and hence, for a cut-free proof of a weak sequent we have $[\![\pi]\!] = \mathrm{H}(\pi)$. We can now lift the main invariance lemma, Lemma 5.4, to proofs of arbitrary end-sequents and formulate this result in terms of the Herbrand-content.

Theorem 7.2. *If $\pi \rightsquigarrow \pi'$ is a reduction sequence of simple proofs, then $[\![\pi]\!] \geqslant [\![\pi']\!]$. If $\pi \overset{ne}{\rightsquigarrow} \pi'$ is a reduction sequence of simple proofs, then $[\![\pi]\!] = [\![\pi']\!]$.*

Proof. If $\pi \rightsquigarrow \pi'$ then $\mathrm{sk}(\pi) \rightsquigarrow \mathrm{sk}(\pi')$ by Lemma 6.9. So, by Lemma 5.4, we have $L(\mathrm{G}(\mathrm{sk}(\pi))) \geqslant L(\mathrm{G}(\mathrm{sk}(\pi')))$. By Lemma 6.10, we get $\mathrm{sk}(L(\mathrm{G}(\pi))) \geqslant \mathrm{sk}(L(\mathrm{G}(\pi')))$. Using Lemma 6.15 and the observation that sk^{-1} commutes with $\leqslant$ we see that

$$[\![\pi]\!] = L(\mathrm{G}(\pi))\rho_{\mathrm{c}} = \mathrm{sk}^{-1}(\mathrm{sk}(L(\mathrm{G}(\pi)))) \geqslant \mathrm{sk}^{-1}(\mathrm{sk}(L(\mathrm{G}(\pi')))) = L(\mathrm{G}(\pi'))\rho_{\mathrm{c}} = [\![\pi']\!]$$

The proof for $\pi \overset{ne}{\rightsquigarrow} \pi'$ is step-by-step the same, replacing $\geqslant$ by $=$. □

Corollary 7.3. *If $\pi \rightsquigarrow \pi'$ is a reduction sequence of simple proofs and π' is cut-free, then*

$$\mathrm{H}(\pi')\rho_{\mathrm{c}} \leqslant [\![\pi]\!] \quad .$$

Proof. This is a direct consequence of Theorem 7.2. □

This corollary shows that $[\![\pi]\!]$ is an upper bound on the Herbrand-disjunctions obtainable by cut-elimination from π. Let us now compare this result with another upper bound that has previously been obtained in [Het10]. To that aim let $\mathrm{G}_0(\pi)$ denote the regular tree grammar underlying $\mathrm{G}(\pi)$ which can be obtained by setting all non-terminals to non-rigid. In this notation, a central result of [Het10], adapted to this paper's setting is

Theorem 7.4. *Let π be a proof of a formula of the shape $\exists x_1 \ldots \exists x_n\, A$ with A quantifier-free, and let $\pi \rightsquigarrow \pi'$ with π' cut-free. Then $\mathrm{H}(\pi') \subseteq L(\mathrm{G}_0(\pi))$.*

While the Theorem 7.4 applies also to non-simple proofs, Corollary 7.3 is stronger in several respects:

First, the size of the Herbrand-content is by an exponential smaller than the size of the bound given by Theorem 7.4. Indeed, it is a straightforward consequence of Lemma 3.9

that the language of a totally rigid acyclic tree grammar with n production rules is bound by n^n but on the other hand:

Proposition 7.5. *There is an acyclic regular tree grammar G with $2n$ productions and $|L(G)| = n^{n^n}$.*

Proof. Let f be an n-ary function symbol, then the productions $\alpha_0 \to f(\alpha_1, \ldots, \alpha_1)$, $\ldots$, $\alpha_{n-1} \to f(\alpha_n, \ldots, \alpha_n)$ create a tree with n^n leaves. Let $c_1, \ldots, c_n$ be terminal symbols, then by adding the productions $\alpha_n \to c_1, \ldots, \alpha_n \to c_n$ we obtain the desired grammar G. □

Secondly, the class of totally rigid acyclic tree grammars can be shown to be in exact correspondence with the class of simple proofs in the following sense. Not only can we use a totally rigid acyclic tree grammar to simulate the process of cut-elimination, we can also—in the other direction—use cut-elimination to simulate the process of calculating the language of a grammar. It is shown in [Het12a] how to transform an arbitrary acyclic totally rigid tree grammar G into a simple proof that has a $\rightsquigarrow$ normal form whose Herbrand-disjunction is essentially the language of G.

The third and—for the purposes of this paper—most important difference is that the bound of Corollary 7.3 is *tight* in the sense that it can actually be reached by a cut-elimination strategy, namely $\overset{ne}{\rightsquigarrow}$. In fact, an even stronger statement is true: not only is there a normal form of $\overset{ne}{\rightsquigarrow}$ that reaches the bound but all of them do. This property leads naturally to the following confluence result for classical logic.

Definition 7.6 (Herbrand-confluence)**.** A relation $\longrightarrow$ on a set of proofs is called *Herbrand-confluent* if $\pi \longrightarrow \pi_1$ and $\pi \longrightarrow \pi_2$ with π_1 and π_2 being normal forms for $\longrightarrow$ implies that $\mathrm{H}(\pi_1)\rho_{\mathrm{c}} = \mathrm{H}(\pi_2)\rho_{\mathrm{c}}$.

Corollary 7.7. *The relation $\overset{ne}{\rightsquigarrow}$ is Herbrand-confluent on the set of simple proofs.*

Proof. This is a direct consequence of Theorem 7.2. □

How does this result fit together with $\overset{ne}{\rightsquigarrow}$ being neither confluent nor strongly normalizing? In fact, note that it is possible to construct a simple proof which permits an infinite $\overset{ne}{\rightsquigarrow}$ reduction sequence from which one can obtain normal forms of arbitrary size by bailing out from time to time. This can be done by building on the propositional double-contraction example found e.g. in [DJS97, Gal93, Urb00] and in a similar form in [Zuc74]. While these infinitely many normal forms do have pairwise different Herbrand-disjunctions when regarded as *multisets*, Corollary 7.7 shows that as *sets* they are all the same. This set-character of Herbrand-disjunctions is assured by using canonical variable names (or equivalently: Skolemization) and thus identifying repeated instances. This observation shows that the lack of strong normalization is taken care of by using sets instead of multisets as data structure. But what about the lack of confluence? Results like [BH11] and [Het12b] show that the number of $\rightsquigarrow$ normal forms with different Herbrand-disjunctions can be enormous. On the other hand we have just seen that $\overset{ne}{\rightsquigarrow}$ induces only *a single* Herbrand-disjunction: $[\![\pi]\!]$. The relation between $[\![\pi]\!]$ and the many Herbrand-disjunctions induced by $\rightsquigarrow$ is explained by Corollary 7.3: $[\![\pi]\!]$ contains them all.

8. Conclusion

We have shown that non-erasing cut-elimination for the class of simple proofs is Herbrand-confluent. While there are different and possibly infinitely many normal forms, they all induce the same Herbrand-disjunction. This result motivates the definition of this unique Herbrand-disjunction as Herbrand-*content* of the proof with cut.

As future work, the authors plan to extend this result to arbitrary first-order proofs. The treatment of blocks of quantifiers is straightforward: the rigidity condition must be changed to apply to vectors of non-terminals. Treating quantifier alternations is more difficult: the current results suggest to use a *stack* of totally rigid tree grammars, each layer of which corresponds to one layer of quantifiers (and is hence acyclic). Concerning further generalizations, note that the method of describing a cut-free proof by a tree language is applicable to any proof system with quantifiers that has a Herbrand-like theorem, e.g., even full higher-order logic as in [Mil87]. The difficulty consists in finding an appropriate type of grammars.

Given the wealth of different methods for the extraction of constructive content from classical proofs, what we learn from our work about the class of simple proofs is this: the first-order structure possesses (in contrast to the propositional structure) a unique and canonical unfolding. The various extraction methods hence do not differ in the choice of how to unfold the first-order structure but only in choosing *which part* of it to unfold. We therefore see that the effect of the underspecification of algorithmic detail in classical logic is redundancy.

Acknowledgments

The authors would like to thank Paul-André Melliès for helpful comments on this work. The first author was supported by a Marie Curie Intra European Fellowship within the 7th European Community Framework Programme, by the projects I603, P22028 and P25160 of the Austrian Science Fund (FWF) and the WWTF Vienna Research Group 12-04.

References

[Avi10] Jeremy Avigad. The computational content of classical arithmetic. In Solomon Feferman and Wilfried Sieg, editors, *Proofs, Categories, and Computations: Essays in Honor of Grigori Mints*, pages 15–30. College Publications, 2010.

[Bar84] Hendrik Pieter Barendregt. *The Lambda Calculus*, volume 103 of *Studies in Logic and the Foundations of Mathematics*. Elsevier, 1984.

[BB96] Franco Barbanera and Stefano Berardi. A Symmetric Lambda Calculus for Classical Program Extraction. *Information and Computation*, 125(2):103–117, 1996.

[BBS02] Ulrich Berger, Wilfried Buchholz, and Helmut Schwichtenberg. Refined Program Extraction from Classical Proofs. *Annals of Pure and Applied Logic*, 114:3–25, 2002.

[BH11] Matthias Baaz and Stefan Hetzl. On the non-confluence of cut-elimination. *Journal of Symbolic Logic*, 76(1):313–340, 2011.

[BHL+05] Matthias Baaz, Stefan Hetzl, Alexander Leitsch, Clemens Richter, and Hendrik Spohr. Cut-Elimination: Experiments with CERES. In Franz Baader and Andrei Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning (LPAR) 2004*, volume 3452 of *Lecture Notes in Computer Science*, pages 481–495. Springer, 2005.

[BHL+08] Matthias Baaz, Stefan Hetzl, Alexander Leitsch, Clemens Richter, and Hendrik Spohr. CERES: An Analysis of Fürstenberg's Proof of the Infinity of Primes. *Theoretical Computer Science*, 403(2–3):160–175, 2008.

[BHW12] Matthias Baaz, Stefan Hetzl, and Daniel Weller. On the complexity of proof deskolemization. *Journal of Symbolic Logic*, 77(2):669–686, 2012.

[BL94] Matthias Baaz and Alexander Leitsch. On Skolemization and Proof Complexity. *Fundamenta Informaticae*, 20(4):353–379, 1994.

[BL00] Matthias Baaz and Alexander Leitsch. Cut-elimination and Redundancy-elimination by Resolution. *Journal of Symbolic Computation*, 29(2):149–176, 2000.

[Bus95] Samuel R. Buss. On Herbrand's Theorem. In *Logic and Computational Complexity*, volume 960 of *Lecture Notes in Computer Science*, pages 195–209. Springer, 1995.

[CDG+07] H. Comon, M. Dauchet, R. Gilleron, C. Löding, F. Jacquemard, D. Lugiez, S. Tison, and M. Tommasi. Tree Automata: Techniques and Applications. Available on: `http://www.grappa.univ-lille3.fr/tata`, 2007. release October, 12th 2007.

[CH00] Pierre-Louis Curien and Hugo Herbelin. The Duality of Computation. In *Proceedings of the Fifth ACM SIGPLAN International Conference on Functional Programming (ICFP '00)*, pages 233–243. ACM, 2000.

[CHM12a] Kaustuv Chaudhuri, Stefan Hetzl, and Dale Miller. A Systematic Approach to Canonicity in the Classical Sequent Calculus. In Patrick Cégielski and Arnaud Durand, editors, *Computer Science Logic (CSL) 2012*, volume 16 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 183–197. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2012.

[CHM12b] Kaustuv Chaudhuri, Stefan Hetzl, and Dale Miller. The Isomorphism Between Expansion Proofs and Multi-Focused Sequent Proofs. submitted, 2012.

[DJS97] Vincent Danos, Jean-Baptiste Joinet, and Harold Schellinx. A New Deconstructive Logic: Linear Logic. *Journal of Symbolic Logic*, 62(3):755–807, 1997.

[Gal93] Jean Gallier. Constructive Logics. Part I: A Tutorial on Proof Systems and Typed λ-Calculi. *Theoretical Computer Science*, 110(2):249–339, 1993.

[Gen38] Gerhard Gentzen. Neue Fassung des Widerspruchsfreiheitsbeweises fur die reine Zahlentheorie. *Forschungen zur Logik und zur Grundlegung der exakten Wissenschaften*, 4:19–44, 1938.

[GS97] Ferenc Gécseg and Magnus Steinby. Tree Languages. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages: Volume 3: Beyond Words*, pages 1–68. Springer, 1997.

[HB39] David Hilbert and Paul Bernays. *Grundlagen der Mathematik II*. Springer, 1939.

[Hei10] Willem Heijltjes. Classical proof forestry. *Annals of Pure and Applied Logic*, 161(11):1346–1366, 2010.

[Her30] Jacques Herbrand. *Recherches sur la théorie de la démonstration*. PhD thesis, Université de Paris, 1930.

[Het10] Stefan Hetzl. On the form of witness terms. *Archive for Mathematical Logic*, 49(5):529–554, 2010.

[Het12a] Stefan Hetzl. Applying Tree Languages in Proof Theory. In Adrian-Horia Dediu and Carlos Martín-Vide, editors, *Language and Automata Theory and Applications (LATA) 2012*, volume 7183 of *Lecture Notes in Computer Science*. Springer, 2012.

[Het12b] Stefan Hetzl. The Computational Content of Arithmetical Proofs. *Notre Dame Journal of Formal Logic*, 53(3):289–296, 2012.

[HLRW13] Stefan Hetzl, Alexander Leitsch, Giselle Reis, and Daniel Weller. Algorithmic Introduction of Quantified Cuts. submitted, 2013.

[HLW12] Stefan Hetzl, Alexander Leitsch, and Daniel Weller. Towards Algorithmic Cut-Introduction. In *Logic for Programming, Artificial Intelligence and Reasoning (LPAR-18)*, volume 7180 of *Lecture Notes in Computer Science*, pages 228–242. Springer, 2012.

[HS12] Stefan Hetzl and Lutz Straßburger. Herbrand-Confluence for Cut-Elimination in Classical First-Order Logic. In Patrick Cégielski and Arnaud Durand, editors, *Computer Science Logic (CSL) 2012*, volume 16 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 320–334. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2012.

[JKV09] Florent Jacquemard, Francis Klay, and Camille Vacher. Rigid tree automata. In Adrian Horia Dediu, Armand-Mihai Ionescu, and Carlos Martín-Vide, editors, *Third International Conference on Language and Automata Theory and Applications (LATA) 2009*, volume 5457 of *Lecture Notes in Computer Science*, pages 446–457. Springer, 2009.

[JKV11] Florent Jacquemard, Francis Klay, and Camille Vacher. Rigid tree automata and applications. *Information and Computation*, 209:486–512, 2011.

[Koh08] Ulrich Kohlenbach. *Applied Proof Theory: Proof Interpretations and their Use in Mathematics*. Springer, 2008.

[McK13] Richard McKinley. Proof nets for Herbrand's Theorem. *ACM Transactions on Computational Logic*, 14(1), 2013.

[Mil87] Dale Miller. A Compact Representation of Proofs. *Studia Logica*, 46(4):347–370, 1987.

[Par92] Michel Parigot. $\lambda\mu$-Calculus: An Algorithmic Interpretation of Classical Natural Deduction. In Andrei Voronkov, editor, *Logic Programming and Automated Reasoning (LPAR) 1992*, volume 624 of *Lecture Notes in Computer Science*, pages 190–201. Springer, 1992.

[RT12] Diana Ratiu and Trifon Trifonov. Exploring the Computational Content of the Infinite Pigeonhole Principle. *Journal of Logic and Computation*, 22(2):329–350, 2012.

[Sch77] Helmut Schwichtenberg. Proof Theory: Some Applications of Cut-Elimination. In J. Barwise, editor, *Handbook of Mathematical Logic*, pages 867–895. North-Holland, 1977.

[UB00] Christian Urban and Gavin Bierman. Strong Normalization of Cut-Elimination in Classical Logic. *Fundamenta Informaticae*, 45:123–155, 2000.

[Urb00] Christian Urban. *Classical Logic and Computation*. PhD thesis, University of Cambridge, October 2000.

[Wel11] Daniel Weller. On the Elimination of Quantifier-Free Cuts. *Theoretical Computer Science*, 412(49):6843–6854, 2011.

[Zuc74] J. Zucker. The Correspondence Between Cut-Elimination and Normalization. *Annals of Mathematical Logic*, 7:1–112, 1974.

This work is licensed under the Creative Commons Attribution-NoDerivs License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nd/2.0/ or send a letter to Creative Commons, 171 Second St, Suite 300, San Francisco, CA 94105, USA, or Eisenacher Strasse 2, 10777 Berlin, Germany