Tree Grammars for the Elimination of Non-prenex Cuts*

Stefan Hetzl and Sebastian Zivota

Institute of Discrete Mathematics and Geometry
Vienna University of Technology
Wiedner Hauptstraße 8-10, 1040 Vienna, Austria

Abstract

Recently a new connection between proof theory and formal language theory was introduced. It was shown that the operation of cut elimination for proofs with prenex Π_1-cuts in classical first-order logic corresponds to computing the language of a particular type of tree grammars. The present paper extends this connection to arbitrary (i.e. non-prenex) cuts without quantifier alternations. The key to treating non-prenex cuts lies in using a new class of tree grammars, constraint grammars, which describe the relationship of the applicability of its productions by a propositional formula.

1998 ACM Subject Classification F.4.1 Mathematical Logic, F.4.2 Grammars and Other Rewriting Systems

Keywords and phrases proof theory, cut-elimination, Herbrand's theorem, tree grammars

Digital Object Identifier 10.4230/LIPIcs.CSL.2015.110

1 Introduction

The constructive content of proofs has always been a central topic of proof theory. A helpful perspective on the constructive content of proofs in classical first-order logic is provided by Herbrand's theorem [7] (see also [2]). It states that from a valid first order formula one can obtain a quantifier-free tautology by expanding existential quantifiers to finite disjunctions of instances and universal quantifiers to finite conjunctions of instances. Provided one is willing to speak about provability instead of validity this result even extends to higher-order logic, see e.g. [17].

It is straightforward to read off a Herbrand expansion from a cut-free proof. On the other hand, proofs with cut can be non-elementarily shorter than the shortest Herbrand expansion [20, 18, 19]. Therefore, in order to compute a Herbrand expansion from a proof with cut, cut-elimination (or another equivalent normalization process) is necessary.

This paper is part of a line of research which was started in [8] and is dedicated to applying methods from formal language theory in proof theory. In [8] a class of tree grammars has been introduced which describe the Herbrand expansion obtained from proofs with prenex Π_1-cuts. The size of the grammar is bound by the size of the proof from which it is read off. The language of the grammar is a Herbrand expansion of size exponential in the size of the grammar. Thus by computing the language of this grammar, the cumbersome computational process of cut-elimination can be circumvented. These grammars owe their simplicity to the fact that they fully abstract from the propositional structure of the proof by speaking

* Supported by the Vienna Science and Technology Fund (WWTF) project VRG12-04 and by the Austrian Science Fund (FWF) project P25160.

24th EACSL Annual Conference on Computer Science Logic (CSL 2015).
Editor: Stephan Kreutzer; pp. 110–127
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

only about witness terms. There are other formalisms which allow to compute a Herbrand expansion in a way that abstracts from the propositional structure: the historically first such formalism is Hilbert's ε-calculus [14]. In [5] Gerhardy and Kohlenbach adapt Shoenfield's variant of Gödel's Dialectica interpretation to a system of pure predicate logic. Recent work, more similar to proof nets, is that of Heijltjes [6] and McKinley [16]. An approach similar to [6, 16] in the formalism of expansion trees [17] can be found in [13].

What sets the grammars introduced in [8] and treated in the present paper apart from the above-mentioned formalisms is that they do not only allow to compute a Herbrand expansion but provide a (well-understood) abstract description of its structure. On the one hand this has the consequence that problems from formal language theory such as membership, inclusion, etc. assume a proof-theoretic meaning and hence standard algorithms can be used for solving the corresponding proof-theoretic problems, usually with smaller asymptotic complexity than the naive algorithms which rely on computing the normal form(s), see e.g. [15]. On the other hand, the strong grip on the structure of a Herbrand expansion afforded by a formal grammar opens the door to the following interesting theoretical and applied investigations:

From strengthening the result of [8] one can show that all (infinitely many) normal forms of the non-erasing Gentzen reduction lead to the same Herbrand expansion, see [12]. This property has been called Herbrand-confluence in [12]. Grammars have been used for a cut-introduction algorithm in [11, 10]. This algorithm has been implemented and empirically evaluated with good results in [9] and it has recently been extended to induction in [4]. In [3] an incompressible sequence of word languages is constructed which via the result of [8] yields a sequence of first-order formulas all of whose cut-free proofs are essentially incompressible by Π_1-cuts.

All of the results so far are limited to prenex Π_1 cuts (with the exception of [1] which treats prenex Π_2 cuts) and consequently all of the applications mentioned above are so as well. In this paper, which is a generalization and an improved presentation of the results obtained in [21], we remove the limitation to prenex formulas by employing a more general class of grammars. This opens the way for extending the results and techniques of [12, 11, 10, 9, 4, 3, 1] to non-prenex cuts and induction formulas.

2 Previous Work

In this paper we will use the proof system **LK** which was introduced by Gentzen in the 1930s. It is a sequent calculus, which means that unlike most other proof systems, derivations do not operate directly on formulas, but rather on so-called sequents. A sequent is a structure of the form $\Gamma \vdash \Delta$, where Γ and Δ are multisets of formulas, respectively called the antecedent and succedent of the sequent. The natural interpretation of $\Gamma \vdash \Delta$ is "the conjunction over Γ implies the disjunction over Δ".

We shall now define the inference rules of **LK** as used in this paper. They are easily seen to be sound given the above interpretation.

▶ **Definition 1** (Rules of **LK**).

1. Axioms: $\overline{A \vdash A}$ with A atomic.
2. Contraction:

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

3. Weakening:

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

4. Propositional rules:

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

$$\frac{A, B, \Gamma \vdash \Delta}{A \wedge B, \Gamma \vdash \Delta}\, \wedge_l \qquad\qquad \frac{\Gamma \vdash \Delta, A \quad \Pi \vdash \Lambda, B}{\Gamma, \Pi \vdash \Delta, \Lambda, A \wedge B}\, \wedge_r$$

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

5. Quantifier rules:

$$\frac{A[x \setminus t], \Gamma \vdash \Delta}{\forall x A, \Gamma \vdash \Delta}\, \forall_l \qquad\qquad \frac{\Gamma \vdash \Delta, A[x \setminus \alpha]}{\Gamma \vdash \Delta, \forall x A}\, \forall_r$$

$$\frac{A[x \setminus \alpha], \Gamma \vdash \Delta}{\exists x A, \Gamma \vdash \Delta}\, \exists_l \qquad\qquad \frac{\Gamma \vdash \Delta, A[x \setminus t]}{\Gamma \vdash \Delta, \forall x A}\, \exists_r$$

Here, t is any term, while α is a variable that does not occur in Γ, Δ or A, called an *eigenvariable*. The inferences that use eigenvariables are called *strong quantifier inferences*, the others *weak quantifier inferences*.

6. The cut rule:

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

The formula A is called the *cut formula* of the inference. We call a cut quantified if its cut formula contains quantifiers. In the sequel, we will refer to the set of quantified cuts in a proof π as $\mathrm{QCuts}(\pi)$.

In all of these cases, the sequents above the line are called premises and the one below is called the conclusion. Additionally, the emphasized formulas in the premises are called auxiliary formulas, while the emphasized formula in the conclusion is called the main or principal formula. Note that some rules (e.g. weakening) do not have auxiliary formulas and the cut rule does not have a main formula.

Often a given formula will occur several times in a proof and these occurrences have different properties. We will mark important formula occurrences like this: $A_{[\mu]}, B_{[\nu]}$, etc.

We can formalize the notion of a formula occurrence being an ancestor of another: μ is an immediate ancestor of ν if there is an inference such that μ is its auxiliary formula and ν its main formula. The "ancestor" relation is then simply the transitive closure of the "immediate ancestor" relation. When we say that a formula is an "ancestor of the end sequent", we mean "ancestor of a formula in the end sequent".

We often visualize proofs as two-dimensional structures with axioms at the top and the conclusion at the bottom. In this context, it makes sense to say that an inference is "above" or "below" another or to talk about "left" and "right" subproofs. We also generally regard proofs as being constructed top-down, so we say for instance that the weakening rule "introduces" a formula.

Gentzen proved that every **LK**-proof can be algorithmically transformed into a cut-free proof, i.e. one that does not contain any cut inferences. The standard proof of cut-elimination in **LK** employs the following set of cut-reduction rules.

▶ **Definition 2** (Cut reduction)**.** Let c be a cut in a proof π and let A_c be the cut formula of c. We define the following steps of cut reduction according to the inferences immediately above the cut:

1. On one side of c, there is a unary or binary inference r whose active formula is not A_c:

$$\dfrac{\begin{matrix}(\psi_1)\\ \Gamma \vdash \Delta, A_c\end{matrix} \quad \dfrac{\begin{matrix}(\psi_2)\\ A_c, \Pi' \vdash \Lambda'\end{matrix}}{A_c, \Pi \vdash \Lambda}\, r}{\Gamma, \Pi \vdash \Delta, \Lambda}\, \mathit{cut} \rightsquigarrow \dfrac{\dfrac{\begin{matrix}(\psi_1)\\ \Gamma \vdash \Delta, A_c\end{matrix} \quad \begin{matrix}(\psi_2)\\ A_c, \Pi' \vdash \Lambda'\end{matrix}}{\Gamma, \Pi' \vdash \Delta, \Lambda'}\, \mathit{cut}}{\Gamma, \Pi \vdash \Delta, \Lambda}\, r$$

$$\dfrac{\begin{matrix}(\psi_1)\\ \Gamma \vdash \Delta, A_c\end{matrix} \quad \dfrac{\begin{matrix}(\psi_2)\\ A_c, \Pi_1 \vdash \Lambda_1\end{matrix} \quad \begin{matrix}(\psi_2)\\ \Pi_2 \vdash \Lambda_2\end{matrix}}{A_c, \Pi \vdash \Lambda}\, r}{\Gamma, \Pi \vdash \Delta, \Lambda}\, \mathit{cut} \rightsquigarrow \dfrac{\dfrac{\begin{matrix}(\psi_1)\\ \Gamma \vdash \Delta, A_c\end{matrix} \quad \begin{matrix}(\psi_2)\\ A_c, \Pi_1 \vdash \Lambda_1\end{matrix}}{\Gamma, \Pi_1 \vdash \Delta, \Lambda_1}\, \mathit{cut} \quad \begin{matrix}(\psi_3)\\ \Pi_2 \vdash \Lambda_2\end{matrix}}{\Gamma, \Pi \vdash \Delta, \Lambda}\, r$$

   The case where r is on the left side of c works entirely symmetrically.

2. A_c is introduced by an axiom on one side of c:

$$\dfrac{A_c \vdash A_c \quad \begin{matrix}(\psi)\\ A_c, \Gamma \vdash \Delta\end{matrix}}{A_c, \Gamma \vdash \Delta}\, \mathit{cut} \rightsquigarrow \begin{matrix}(\psi)\\ A_c, \Gamma \vdash \Delta\end{matrix}$$

3. A_c is introduced by a weakening on one side of c:

$$\dfrac{\dfrac{\begin{matrix}(\psi_1)\\ \Gamma \vdash \Delta\end{matrix}}{\Gamma \vdash \Delta, A_c}\, w_r \quad \begin{matrix}(\psi_2)\\ A_c, \Pi \vdash \Lambda\end{matrix}}{\Gamma, \Pi \vdash \Delta, \Lambda}\, \mathit{cut} \rightsquigarrow \dfrac{\begin{matrix}(\psi_1)\\ \Gamma \vdash \Delta\end{matrix}}{\Gamma, \Pi \vdash \Delta, \Lambda}\, w*$$

   The case where the weakening is on the right side is symmetrical.

4. A_c is the main formula of a contraction on one side of c:

$$\dfrac{\dfrac{\begin{matrix}(\psi_1)\\ \Gamma \vdash \Delta, A_c, A_c\end{matrix}}{\Gamma \vdash \Delta, A_c}\, c_r \quad \begin{matrix}(\psi_2)\\ A_c, \Pi \vdash \Lambda\end{matrix}}{\Gamma, \Pi \vdash \Delta, \Lambda}\, \mathit{cut} \rightsquigarrow \dfrac{\dfrac{\dfrac{\begin{matrix}(\psi_1)\\ \Gamma \vdash \Delta, A_c, A_c\end{matrix} \quad \begin{matrix}(\psi_2'')\\ A_c, \Pi \vdash \Lambda\end{matrix}}{\Gamma, \Pi \vdash \Delta, \Lambda, A_c}\, \mathit{cut} \quad \begin{matrix}(\psi_2')\\ A_c, \Pi \vdash \Lambda\end{matrix}}{\Gamma, \Pi, \Pi \vdash \Delta, \Lambda, \Lambda}\, \mathit{cut}}{\Gamma, \Pi \vdash \Delta, \Lambda}\, c*$$

   Here, ψ_2' and ψ_2'' each arise from ψ_2 by replacing all eigenvariables introduced in ψ_2 with fresh copies. The case where the contraction is on the right is treated analogously.

5. $A_c = \exists x B$ and A_c is introduced by $\exists$-inferences immediately above the cut:

$$\dfrac{\dfrac{\begin{matrix}(\psi_1)\\ \Gamma \vdash \Delta, B[x \backslash t]\end{matrix}}{\Gamma \vdash \Delta, \exists x B}\, \exists_r \quad \dfrac{\begin{matrix}(\psi_2)\\ B[x \backslash t], \Pi \vdash \Lambda\end{matrix}}{\exists x B, \Pi \vdash \Lambda}\, \exists_l}{\Gamma, \Pi \vdash \Delta, \Lambda}\, \mathit{cut} \rightsquigarrow \dfrac{\begin{matrix}(\psi_1)\\ \Gamma \vdash \Delta, B[x \backslash t]\end{matrix} \quad \begin{matrix}(\psi_2[\alpha \backslash t])\\ B[x \backslash t], \Pi \vdash \Lambda\end{matrix}}{\Gamma, \Pi \vdash \Delta, \Lambda}\, \mathit{cut}$$

6. $A_c = \forall x B$: Analogous to the previous case, but with switched sides.

7. $A_c = B \wedge C$ and A_c is introduced by $\wedge$-inferences immediately above the cut:

$$\dfrac{\dfrac{\begin{matrix}(\psi_1) \\ \Gamma_1 \vdash \Delta_1, B\end{matrix} \quad \begin{matrix}(\psi_2) \\ \Gamma_2 \vdash \Delta_2, C\end{matrix}}{\Gamma \vdash \Delta, B \wedge C}\,\wedge_r \quad \dfrac{\begin{matrix}(\psi_3) \\ B, C, \Pi \vdash \Lambda\end{matrix}}{B \wedge C, \Pi \vdash \Lambda}\,\wedge_l}{\Gamma, \Pi \vdash \Delta, \Lambda}\,cut \quad \rightsquigarrow$$

$$\rightsquigarrow \quad \dfrac{\begin{matrix}(\psi_1) \\ \Gamma_1 \vdash \Delta_1, B\end{matrix} \quad \dfrac{\begin{matrix}(\psi_2) \\ \Gamma_2 \vdash \Delta_2, C\end{matrix} \quad \begin{matrix}(\psi_3) \\ C, B, \Pi \vdash \Lambda\end{matrix}}{B, \Gamma_2, \Pi \vdash \Delta_2, \Lambda}\,cut}{\Gamma, \Pi \vdash \Delta, \Lambda}\,cut$$

8. $A_c = B \vee C$: Analogous to the previous case.
9. $A_c = \neg B$ and both $\neg$-inferences introducing A_c are immediately above the cut:

$$\dfrac{\dfrac{\begin{matrix}(\psi_1) \\ B, \Gamma \vdash \Delta\end{matrix}}{\Gamma \vdash \Delta, \neg B}\,\neg_r \quad \dfrac{\begin{matrix}(\psi_2) \\ \Pi \vdash \Lambda, B\end{matrix}}{\neg B, \Pi \vdash \Lambda}\,\neg_l}{\Gamma, \Pi \vdash \Delta, \Lambda}\,cut \quad \rightsquigarrow \quad \dfrac{\begin{matrix}(\psi_2) \\ \Pi \vdash \Lambda, B\end{matrix} \quad \begin{matrix}(\psi_1) \\ B, \Gamma \vdash \Delta\end{matrix}}{\Gamma, \Pi \vdash \Delta, \Lambda}\,cut$$

If π' arises from π by finitely many applications of these rules, then we write $\pi \rightsquigarrow^* \pi'$.

It will often be useful to consider signed formulas, i.e. formulas annotated as either occurring in the antecedent or the consequent of a sequent. The former will be written as $A \vdash$, the latter as $\vdash A$.

▶ **Definition 3** (Herbrand set). Let $\mathcal{S} = A_1, \ldots, A_m \vdash B_1, \ldots, B_n$ be a sequent. An *Herbrand set* of $\mathcal{S}$ is a set $\mathcal{H}$ for which the following two conditions hold:
1. $\mathcal{H} = \mathcal{H}^a \dot\cup \mathcal{H}^s$ where
   - Every element of $\mathcal{H}^a$ is of the form $A \vdash$ with A an instance of some A_i
   - Every element of $\mathcal{H}^s$ is of the form $\vdash B$ with B an instance of some B_j
2. Let $\mathcal{H}'$ be the image of $\mathcal{H}$ under the function $\begin{cases} A \vdash & \mapsto \neg A, \\ \vdash B & \mapsto B \end{cases}$. Then $\bigvee \mathcal{H}'$ is a tautology.

For the sake of simplicity, we abbreviate the latter condition as "$\mathcal{H}$ is a tautology".

It is straightforward to extract an Herbrand set of $\mathcal{S}$ from a cut-free proof of $\mathcal{S}$. By extension, one could in principle also extract an Herbrand set from a proof with cuts by performing cut elimination. There is another possibility, however: an Herbrand set can be viewed as a finite tree language by viewing both $\cdot \vdash$ and $\vdash \cdot$ as well as the propositional connectives and the predicate symbols as function symbols. Finite tree languages can be compactly represented by tree grammars. Our aim is to extract a tree grammar from a proof with cuts such that computing the language's grammar corresponds to performing cut reduction on the proof.

It is well-known that cut-elimination leads to a non-elementary increase in proof length [20, 18, 19]. On the other hand, the size of a Herbrand set is closely related to the length of a cut-free proof. Consequently, for this approach to make sense, the tree grammar must be polynomial in the length of the proof with cut. It thus provides a representation of a Herbrand set as compressed as the proof with cut.

▶ **Definition 4** (Regular tree grammar). A *regular tree grammar* is a tuple $G = \langle \varphi, N, \Sigma, P \rangle$, where
1. Σ is a finite ranked alphabet; its elements are called *terminal symbols* (or *terminals* for short);

2. N is a finite set, disjoint from Σ; its elements are called *nonterminals*;
3. $\varphi \in N$ is the *starting symbol*;
4. P is the set of *productions*, i.e. elements of the form $\alpha \to t$ where $\alpha \in N$ and t is a term over $N \cup \Sigma$.

Let G be a regular tree grammar. A *derivation* in G is a finite sequence $d = \langle \varphi = t_0, t_1, \ldots, t_n \rangle$ such that t_i can be obtained from t_{i-1} by application of a production of G; that is, there is a production $\alpha_i \to s_i \in P$ such that replacing one occurrence of α_i in t_{i-1} with s_i yields t_i. We say that t_n can be derived in G.

Now we can define the language $L(G)$ of G: $L(G)$ is the set of terms over Σ that can be derived in G.

▶ **Example 5.** Consider the regular tree grammar $G = \langle \varphi, N, \Sigma, P \rangle$ with

$$N = \{\varphi, x, y\}$$
$$\Sigma = \{a/0, b/0, g/1, f/2\}$$
$$P = \{\varphi \to f(x, y), x \to a|g(y), y \to a|b\}.$$

The language of G is $\{f(a, a), f(a, b), f(g(a), a), f(g(a), b), f(g(b), a), f(g(b), b)\}$.

▶ **Definition 6.** A totally rigid tree grammar is a regular tree grammar $G = \langle \varphi, N, \Sigma, P \rangle$ with an additional restriction on derivations. Let d be a derivation of G in the sense of regular tree grammars. Then d is a derivation of the totally rigid grammar G if for each $\alpha \in N$, at most one production beginning with α is used in d.

▶ **Example 7.** Let G be the grammar from Example 5. If we view it as a totally rigid grammar, its language is reduced to $\{f(a, a), f(a, b), f(g(a), a), f(g(b), b)\}$.

It is easy to see that the language of a totally rigid grammar is always finite.

The following theorem was proved in [8].

▶ **Theorem 8.** *Let π be a proof with the following properties:*
1. *The end sequent of π is of the form $\vdash \exists \bar{x} A$ with A quantifier-free;*
2. *All cut formulas in π are of the form $\exists y B$ with B quantifier-free.*

Then there is a totally rigid tree grammar $G(\pi)$ such that $L(G(\pi))$ is an Herbrand set of $\exists \bar{x} A$. Moreover, if $|G|$ is the number of productions of G and $|\pi|$ the number of inferences in π, then $|G| \leq |\pi|$.

In this paper we will generalize this result to non-prenex cut formulas and arbitrary end sequents. In order to do that, we will need more powerful grammars.

3 Constraint grammars

Totally rigid grammars are obtained from regular tree grammars by placing restrictions on how many productions can be used per nonterminal in a derivation. Similarly, constraint grammars allow us to restrict which combinations of productions of different nonterminals can be used. This is essential if we want to deal with cut formulas that are non-prenex and contain more than one quantifier.

▶ **Definition 9** (Constraint grammar). A constraint grammar is a tuple $G = \langle \varphi, N, \Sigma, P, \mathcal{C} \rangle$ consisting of a totally rigid grammar $G' = \langle \varphi, N, \Sigma, P \rangle$ together with a *constraint formula* $\mathcal{C}$, which is a propositional formula that uses the productions in P as atoms.

When writing constraint formulas, we will use the symbol "$\rightarrow$" to denote productions and "$\Rightarrow$" for implications.

Any derivation d of the underlying totally rigid grammar of G induces an interpretation v_d of $\mathcal{C}$ in the following manner: If $\alpha \in N$ such that α does not occur in d, then $v_d(p) = \top$ for all $p \in P_\alpha$. If α occurs in d, v_d evaluates the α-productions used in d as $\top$ and the others as $\bot$. This leads to the definition of a valid derivation of G: d is valid iff $v_d(\mathcal{C}) = \top$ (i.e. v_d is a model of $\mathcal{C}$).

A term over Σ is derivable in G if it is derivable in G' via a valid derivation.

Note that determining whether a given derivation is valid for G can be done in linear time relative to the size of d and $\mathcal{C}$.

▶ **Example 10.** Let G be the totally rigid grammar from Example 7. If we extend it to a constraint grammar G' by adding the constraint formula $\mathcal{C} := x \rightarrow a \vee y \rightarrow a$, then $L(G') = \{f(a,a), f(a,b), f(g(a),a)\}$.

4 The grammar of a proof

In this section we will give the central definition of this paper: the constraint grammar induced by a proof.

When working in sequent calculus, it is customary to distinguish between weak and strong quantifiers. Briefly, a quantifier is said to be "strong" if it is universal and below an even number of negations or existential and below an odd number of negations. Conversely, it is called "weak" if it is universal and below an odd number of negations or existential and below an even number. Note that in this context, both the left side of an implication and the antecedent of a sequent count as one negation each.

In the sequel, we always place some restrictions on the proofs we consider.

- The names of bound variables in the end sequent are distinct. This can always be ensured via renaming.
- There are no strong quantifiers in the end sequent. This assumption is justified because we can perform validity-preserving Skolemization, i.e. replace all strong quantifiers by Skolem symbols.
- Each cut formula contains only weak or strong quantifiers, but not both. We call a cut formula Σ_1 or Π_1 accordingly.

From now on, we will call proofs with these properties *simple*.

The above restriction on cut formulas allows us to define the "weak side" and the "strong side" of a cut: Let A_c be the cut formula of a cut c and assume that A_c contains quantifiers. If A_c is Σ_1, each quantifier in c is introduced via a weak quantifier inference in the left subproof of c and a strong inference in the right subproof. Consequently, the left and right subproof are called the "weak" and "strong" side, respectively. In the case of a Π_1 cut formula, the sides are switched. Each quantifier in A_c may be introduced several times on both the weak and the strong side of c; this happens via eigenvariables on the the strong side and arbitrary terms on the weak side. We refer to those eigenvariables and terms as "belonging to" or "being associated with" the quantifier.

We shall now define a constraint grammar $G(\pi) = \langle \varphi, N(\pi), \Sigma, P(\pi), \mathcal{C}(\pi)\rangle$ piece by piece.

▶ **Definition 11** (Terminals and nonterminals of $G(\pi)$)**.** Let π be a simple proof of $A_1, \dots, A_m \vdash B_1, \dots, B_n$.

- Terminals: The terminal symbols Σ of $G(\pi)$ consist of the language of π together with a new symbol w. w will be used to mark places where a formula is introduced by weakening in π.
- Nonterminals: We define sets $\mathrm{N_{ES}}(\pi)$ and $\mathrm{N_{Cuts}}(\pi)$. Let $BV(A)$ be the set of bound variables in the formula A and φ a new symbol. Then $\mathrm{N_{ES}}(\pi) = \{\varphi\} \cup \bigcup_{i=1}^{m} BV(A_i) \cup \bigcup_{i=1}^{n} BV(B_i)$. Since there are no strong quantifiers in the end sequent of π, all strong quantifier inferences must act on ancestors of cut formulas. Thus each eigenvariable is uniquely associated with a particular cut. We write $EV(c)$ for the eigenvariables associated with cut c and $EV(\pi)$ for all eigenvariables in π. This leads to $\mathrm{N_{Cuts}}(\pi) = EV(\pi)$.

Finally, $N(\pi) := \mathrm{N_{ES}}(\pi) \cup \mathrm{N_{Cuts}}(\pi)$.

For any formula A, let $\widehat{A}$ be the matrix of A, i.e. the formula that results from deleting all quantifiers from A.

▶ **Definition 12** (Productions of $G(\pi)$)**.** Let π be a simple proof of $A_1, \dots, A_m \vdash B_1, \dots, B_n$. We define sets $\mathrm{P_{ES}}(\pi)$ and $\mathrm{P_{Cuts}}(\pi)$:

For $i = 1, \dots, m$, $(\varphi \to \widehat{A_i} \vdash) \in \mathrm{P_{ES}}(\pi)$. For $j = 1, \dots, n$, $(\varphi \to \vdash \widehat{B_j}) \in \mathrm{P_{ES}}(\pi)$.

For $x \in \mathrm{N_{ES}}(\pi)$, if π contains an inference $\dfrac{\Gamma \vdash \Delta, A[x \setminus t]}{\Gamma \vdash \Delta, \exists x A}\ \exists_r$, then $x \to t \in \mathrm{P_{ES}}(\pi)$. Moreover, if x is introduced by weakening at least once in π, then $x \to w \in \mathrm{P_{ES}}(\pi)$.

Let $\alpha \in \mathrm{N_{Cuts}}(\pi)$, then α is used to introduce a strong quantifier on some variable z in a cut formula. If the weak side of the cut contains an inference $\dfrac{\Gamma \vdash \Delta, B[z \setminus s]}{\Gamma \vdash \Delta, \exists z B}\ \exists_r$, then $z \to s \in \mathrm{P_{Cuts}}(\pi)$. Moreover, if z is introduced by weakening at least once on the weak side of the cut, then $z \to w \in \mathrm{P_{Cuts}}(\pi)$. $P(\pi) := \mathrm{P_{ES}}(\pi) \cup \mathrm{P_{Cuts}}(\pi)$.

▶ **Definition 13** (Constraint formula of $G(\pi)$)**.** Let π be a simple proof and μ any formula occurrence in π. We define a formula $q(\mu, \pi)$ by induction:

- If μ is quantifier-free, then $q(\mu, \pi) := \top$.
- If μ is introduced by a weakening, then let $z_1, \dots, z_k$ be the weakly bound variables in μ. There are two cases to consider. If μ is ancestor of a cut formula, then for each i let $\alpha_{i,1}, \dots \alpha_{i,n_i}$ be the eigenvariables used to introduce the quantifier over z_i on the strong side and $q(\mu, \pi) := \bigwedge_{i=1}^{k} \bigwedge_{j=1}^{n_i} \alpha_{i,j} \to w$. If μ is ancestor of a formula in the end sequent, then $q(\mu, \pi) := \bigwedge_{i=1}^{k} z_i \to w$.
- If μ is introduced by a quantifier rule, i.e. $\dfrac{\Gamma \vdash \Delta, (A[x \setminus t])_{[\mu']}}{\Gamma \vdash \Delta, (\exists x A)_{[\mu]}}\ \exists_r$, then we make a similar case distinction as in the previous case. If μ is ancestor of a cut formula, then let $\alpha_1, \dots, \alpha_n$ be the eigenvariables of the quantifier of x on the strong side of the cut and $q(\mu, \pi) := \left(\bigvee_{j=1}^{n} \alpha_j \to t\right) \wedge q(\mu', \pi)$. Otherwise, $q(\mu, \pi) := x \to t \wedge q(\mu', \pi)$. The case of a $\forall_l$-inference is analogous.
- If μ is introduced by a $\wedge_r$-rule, as in $\dfrac{\Gamma_1 \vdash \Delta_1, A_{[\nu_1]} \quad \Gamma_2 \vdash \Delta_2, B_{[\nu_2]}}{\Gamma \vdash \Delta, (A \wedge B)_{[\mu]}}\ \wedge_r$, then $q(\mu, \pi) := q(\nu_1, \pi) \wedge q(\nu_2, \pi)$. An $\vee_l$-inference is treated analogously.
- If μ is introduced by a $\wedge_l$-rule, as in $\dfrac{A_{[\nu_1]}, B_{[\nu_2]}, \Gamma \vdash \Delta}{(A \wedge B)_{[\mu]}, \Gamma \vdash \Delta}\ \wedge_l$, then $q(\mu, \pi) := q(\nu_1, \pi) \wedge q(\nu_2, \pi)$, and analogously for $\vee_r$.
- If μ arises from a contraction on the right, i.e. $\dfrac{\Gamma \vdash \Delta, A_{[\nu_1]}, A_{[\nu_2]}}{\Gamma \vdash \Delta, A_{[\mu]}}\ c_r$, then $q(\mu, \pi) := q(\nu_1, \pi) \vee q(\nu_2, \pi)$, and analogously for a contraction on the left.

- If μ is introduced by a $\neg_r$ rule, as in $\dfrac{\Gamma, A_{[\nu]} \vdash \Delta}{\Gamma \vdash \Delta, (\neg A)_{[\mu]}}\ \neg_r$, then $q(\mu, \pi) := q(\nu, \pi)$. A $\neg_l$-inference is treated analogously.
- We skip over all inferences whose active formula is not μ.

Now let A be any formula and $\mu_1, \ldots, \mu_m$ and $\nu_1, \ldots, \nu_n$ the occurrences of A in the antecedent and the succedent of the end sequent, respectively. Then

$$\mathcal{C}_A^{ant}(\pi) := (\varphi \to \widehat{A} \vdash) \Rightarrow \bigvee_{i=1}^{m} q(\mu_i, \pi)$$

$$\mathcal{C}_A^{suc}(\pi) := (\varphi \to \vdash \widehat{A}) \Rightarrow \bigvee_{j=1}^{n} q(\nu_j, \pi)$$

This yields the constraint formula of the end sequent:

$$\mathcal{C}_{\mathrm{ES}}(\pi) := \bigwedge_{A \in ES(\pi)} (\mathcal{C}_A^{ant}(\pi) \wedge \mathcal{C}_A^{suc}(\pi))$$

Furthermore, let $c \in \mathrm{QCuts}(\pi)$ and μ_0 the weak occurrence of its cut formula. Then

$$\mathcal{C}_c(\pi) := q(\mu_0, \pi).$$

Finally we obtain

$$\mathcal{C}(\pi) := \mathcal{C}_{\mathrm{ES}}(\pi) \wedge \bigwedge_{c \in \mathrm{QCuts}(\pi)} \mathcal{C}_c(\pi), \tag{1}$$

the constraint formula of π.

▶ **Definition 14** (Grammar of a proof)**.** Let π be a simple proof. The constraint grammar $G(\pi) := \langle \varphi, N(\pi), \Sigma(\pi), P(\pi), \mathcal{C}(\pi) \rangle$ is called the grammar of π.

The purpose of $\mathcal{C}(\pi)$ is to describe the set of tuples of instances that actually occur in the proof.

▶ **Example 15.** Let π be the following proof:

$$\dfrac{\dfrac{\begin{matrix}(\pi_1) \\ P(f(a,c)) \vee Q(b) \vdash A_c\end{matrix} \quad \begin{matrix}(\pi_2) \\ A_c \vdash \exists x P(x), \exists y Q(y)\end{matrix}}{P(f(a,c)) \vee Q(b) \vdash \exists x P(x), \exists y Q(y)}\ cut_{[c]}}{P(f(a,c)) \vee Q(b) \vdash \exists x P(x) \vee \exists y Q(y)}\ \vee_r$$

where

$$\pi_1 = \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(f(a,c)) \vdash P(f(a,c))}{P(f(a,c)) \vdash \exists z_2 P(f(a,z_2))}\ \exists_r}{P(f(a,c)) \vdash \exists z_2 P(f(a,z_2)), Q(a)}\ w_r}{P(f(a,c)) \vdash \exists z_2 P(f(a,z_2)) \vee Q(a)}\ \vee_r}{P(f(a,c)) \vdash A_c}\ \exists_r \quad \dfrac{\dfrac{\dfrac{Q(b) \vdash Q(b)}{Q(b) \vdash \exists z_2 P(f(b,z_2)), Q(b)}\ w_r}{Q(b) \vdash \exists z_2 P(f(b,z_2)) \vee Q(b)}\ \vee_r}{Q(b) \vdash A_c}\ \exists_r}{P(f(a,c)) \vee Q(b) \vdash A_c, A_c}\ \vee_l}{P(f(a,c)) \vee Q(b) \vdash A_c}\ c_r$$

$$\pi_2 = \dfrac{\dfrac{\dfrac{\dfrac{P(f(\alpha,\beta)) \vdash P(f(\alpha,\beta))}{P(f(\alpha,\beta)) \vdash \exists x P(x)}\ \exists_r}{\exists z_2 P(f(\alpha,z_2)) \vdash \exists x P(x)}\ \exists_l \quad \dfrac{Q(\alpha) \vdash Q(\alpha)}{Q(\alpha) \vdash \exists y Q(y)}\ \exists_r}{\exists z_2 P(f(\alpha,z_2)) \vee Q(\alpha) \vdash \exists x P(x), \exists y Q(y)}\ \vee_l}{A_c \vdash \exists x P(x), \exists y Q(y)}\ \exists_l$$

Here, A_c is the formula $\exists z_1(\exists z_2 P(f(z_1, z_2)) \vee Q(z_1))$. The various parts of $G(\pi)$ are:

- Terminals: $P/1, Q/1, f/2, a/0, b/0, c/0, w/0$
- Nonterminals: $\varphi, x, y, \alpha, \beta$
- Productions:

$$\begin{aligned}
&\varphi \to P(f(a,c)) \vee Q(b) \vdash \ \mid\ \vdash P(x) \vee Q(y),\\
&x \to f(\alpha, \beta),\\
&y \to \alpha,\\
&\alpha \to a|b,\\
&\beta \to c|w.
\end{aligned}$$

- Constraint formula:

$$\begin{aligned}
\mathcal{C}_{\mathrm{ES}}(\pi) &= ((\varphi \to \vdash P(x) \vee Q(y)) \Rightarrow (x \to f(\alpha,\beta) \wedge y \to \alpha))\\
&\quad \wedge ((\varphi \to P(f(a,c)) \vee Q(b) \vdash) \Rightarrow \top),\\
\mathcal{C}_c(\pi) &= (\alpha \to a \wedge \beta \to c) \vee (\alpha \to b \wedge \beta \to w),\\
\mathcal{C}(\pi) &= \mathcal{C}_{\mathrm{ES}}(\pi) \wedge \mathcal{C}_c(\pi).
\end{aligned}$$

Consequently, the language of $G(\pi)$ is

$$L(G(\pi)) = \{P(f(a,c)) \vee Q(b) \vdash\ ,\ \vdash P(f(a,c)) \vee Q(a)\ ,\ \vdash P(f(b,w)) \vee Q(b)\}$$

The following theorem is the main result of this paper.

▶ **Theorem 16.** *Let π be a simple proof of $\Gamma \vdash \Delta$. Then $L(G(\pi))$ is an Herbrand set of $\Gamma \vdash \Delta$.*

The first step towards proving this result will be to show that $L(G(\pi))$ is a Herbrand set if π is almost cut-free, more precisely: if π contains only cuts without quantifiers.

▶ **Lemma 17.** *Let π be a simple proof of $\Gamma \vdash \Delta$ in which no cut formula contains a quantifier. Then $L(G(\pi))$ is an Herbrand set of $\Gamma \vdash \Delta$.*

Proof. By induction on the length of π. The case of π a one-line proof of an axiom is trivial. Now we consider the various possibilities for the lowest inference of π. We only consider one of the cedents in each case; the other one is treated analogously. Moreover, we only show the validity of the language; the fact that it consists of instances of the end sequent is immediately obvious. Recall that we say "$L(G(\pi))$ is a tautology" to mean "the image of $L(G(\pi))$ under the function $\begin{cases} A \vdash & \mapsto \neg A, \\ \vdash B & \mapsto B \end{cases}$ is a tautology".

- Weakening: Let $\pi = \dfrac{\begin{matrix}(\pi')\\ \Gamma \vdash \Delta\end{matrix}}{\Gamma \vdash \Delta, A}\ w_r$. Let $x_1, \ldots, x_n$ be the bound variables in A. Obviously $L(G(\pi)) = L(G(\pi')) \cup \{\vdash A[x_1 \setminus w, \ldots, x_n \setminus w]\}$ is an Herbrand set if $L(G(\pi'))$ is.
- Contraction: Let $\pi = \dfrac{\begin{matrix}(\pi')\\ \Gamma \vdash \Delta, A_{[\mu_1]}, A_{[\mu_2]}\end{matrix}}{\Gamma \vdash \Delta, A_{[\mu]}}\ c_r$. It is easy to see that the constraint formulas, nonterminals and productions are unchanged between π and π'. Therefore, $L(G(\pi)) = L(G(\pi'))$.

- Negation: Let $\pi = \dfrac{\begin{matrix}(\pi')\\ A_{[\mu']}, \Gamma \vdash \Delta\end{matrix}}{\Gamma \vdash \Delta, (\neg A)_{[\mu]}} \neg_r$. Observe that if

$$\mathcal{C}_A^{ant}(\pi') = (\varphi \to \widehat{A} \vdash) \Rightarrow (q(\mu', \pi') \vee \mathcal{B}_1),$$
$$\mathcal{C}_{\neg A}^{suc}(\pi') = (\varphi \to \vdash \neg\widehat{A}) \Rightarrow \mathcal{B}_2,$$

then

$$\mathcal{C}_A^{ant}(\pi) = (\varphi \to \widehat{A} \vdash) \Rightarrow \mathcal{B}_1,$$
$$\mathcal{C}_{\neg A}^{suc}(\pi) = (\varphi \to \vdash \neg\widehat{A}) \Rightarrow (q(\mu, \pi) \vee \mathcal{B}_2).$$

It follows that $\vdash \neg\widehat{A}$ is derivable in $G(\pi)$ iff $\widehat{A} \vdash$ is derivable in $G(\pi')$. Clearly, $L(G(\pi))$ is an Herbrand set if $L(G(\pi'))$ is.

- Disjunction: Let $\pi = \dfrac{\begin{matrix}(\pi')\\ \Gamma \vdash \Delta, A_{[\mu_1]}, B_{[\mu_2]}\end{matrix}}{\Gamma \vdash \Delta, (A \vee B)_{[\mu]}} \vee_r$. The language of $G(\pi')$ can be written as $L_{\vdash A} \cup L_{\vdash B} \cup L_{\Gamma\vdash\Delta}$, where $L_{\vdash A}$ contains those derivable formulas that are obtained by starting with the production $\varphi \to \vdash \widehat{A}$, and analogously for $L_{\vdash B}$ and $L_{\Gamma\vdash\Delta}$. Note that these sets are not necessarily disjoint; for instance, A and B might coincide or one of them might occur in the context. In $G(\pi)$, the nonterminals and the constraint formula are unchanged, but φ has the production $\varphi \to \vdash \widehat{A} \vee \widehat{B}$. This means that $L(G(\pi)) = L_{\vdash A\vee B} \cup L_{\Gamma\vdash\Delta}$, where $L_{\vdash A\vee B} = \{\vdash A' \vee B' : \vdash A' \in L_A, \vdash B' \in L_B\}$. It follows that if $L(G(\pi'))$ is a tautology, so is $L(G(\pi))$.

- Conjunction: Let $\pi = \dfrac{\begin{matrix}(\pi')\\ \Gamma \vdash \Delta, A_{[\mu_1]}\end{matrix} \quad \begin{matrix}(\pi'')\\ \Pi \vdash \Lambda, B_{[\mu_2]}\end{matrix}}{\Gamma, \Pi \vdash \Delta, \Lambda, (A \wedge B)_{[\mu]}} \wedge_r$. Similarly to the previous case, write

$$L(G(\pi')) = L_{\vdash A} \cup L_{\Gamma\vdash\Delta},$$
$$L(G(\pi'')) = L_{\vdash B} \cup L_{\Pi\vdash\Lambda},$$
$$L(G(\pi)) = L_{\vdash A\wedge B} \cup L_{\Gamma\vdash\Pi} \cup L_{\Delta\vdash\Lambda}$$

Given any interpretation of the atoms in $L(G(\pi))$, there are two possibilities. If any element of $L_{\Gamma\vdash\Pi}$ or $L_{\Delta\vdash\Lambda}$ is true under the interpretation, we are done. If all of them are false, then some $\vdash A' \in L_{\vdash A}$ and $\vdash B' \in L_{\vdash B}$ must be true by induction. This means that $\vdash A' \wedge B' \in L_{\vdash A\wedge B}$ is also true. Thus, $L(G(\pi))$ is a tautology.

- Existential quantifier: Let $\pi = \dfrac{\begin{matrix}(\pi')\\ \Gamma \vdash \Delta(A[x \setminus t])_{[\mu']}\end{matrix}}{\Gamma \vdash \Delta, (\exists x A)_{[\mu]}} \exists_{r[\iota]}$. In $G(\pi)$, the production $\varphi \to \widehat{A}[x \setminus t]$ that exists in $G(\pi')$ is replaced by $\varphi \to \widehat{A}$ and $x \to t$. If $\mathcal{C}(\pi')$ contains the subformulas

$$\mathcal{C}_{A[x \setminus t]}^{suc}(\pi') = (\varphi \to \widehat{A}[x \setminus t]) \Rightarrow (q(\mu', \pi') \vee \mathcal{B}_1),$$
$$\mathcal{C}_{\exists x A}^{suc}(\pi') = (\varphi \to \widehat{A}) \Rightarrow \mathcal{B}_2,$$

then $\mathcal{C}(\pi)$ contains

$$\mathcal{C}_{A[x \setminus t]}^{suc}(\pi) = (\varphi \to \widehat{A}[x \setminus t]) \Rightarrow \mathcal{B}_1,$$
$$\mathcal{C}_{\exists x A}^{suc}(\pi) = (\varphi \to \widehat{A}) \Rightarrow (((x \to t) \wedge q(\mu', \pi)) \vee \mathcal{B}_2).$$

If $\vdash C$ (or $C \vdash$) $\in L(G(\pi'))$ is an instance of a formula in the context, then it can still be derived in $G(\pi)$. If $\vdash C$ is an instance of $A[x \setminus t]$ and d' a derivation leading to it, d' must begin with $\varphi \to \widehat{A}[x \setminus t]$. We can transform d' into a valid derivation d of $G(\pi)$ by replacing this first step with $\varphi \to \widehat{A} \to \widehat{A}[x \setminus t]$. Thus, the two languages coincide.

- Cut: Let

$$\begin{aligned} L(G(\pi')) &= L'_{\vdash A} \cup L_{\Gamma \vdash \Delta}, \\ L(G(\pi'')) &= L''_{A \vdash} \cup L_{\Pi \vdash \Lambda}, \\ L(G(\pi)) &= L_{\Gamma, \Pi \vdash \Delta, \Lambda} = L_{\Gamma, \Pi \vdash} \cup L_{\vdash \Delta, \Lambda} \end{aligned}$$

as before. Our goal is to show that $\bigvee L_{\Gamma, \Pi \vdash \Delta, \Lambda}$ is tautological. First of all, note that the cut formula is quantifier-free and hence its occurrences only contribute one instance each to the languages of their respective grammars, namely respectively $\vdash A$ and $A \vdash$. Now pick any interpretation. If any element of $L_{\Gamma, \Pi \vdash}$ is true, we are done. Otherwise either an element of $L_{\vdash \Delta}$ or A must be true because $L(G(\pi'))$ is a tautology. In the former case we are, again, done; in the latter case, an element of $L(G(\pi''))$ must be true. This element can be neither A itself nor anything in $L_{\vdash \Pi}$, so it must be an element of L_Λ. Thus, under each interpretation, an element of $L(G(\pi))$ evaluates to true. ◀

5 Cut elimination and grammars

▶ **Definition 18** ($\leq$ relation for formulas)**.** We define a relation $\leq$ between formulas: $A \leq B$ if B can be obtained by replacing occurrences of w in A with terms. Note that different occurrences of w may be replaced with different terms. A strict semantic definition of $A \leq B$ can be achieved in the following manner: Let A' be the formula that results from making all the occurrences of w in A distinct, i.e. replacing each w with a new constant symbol w_i. Then $\forall \bar{w} A' \Rightarrow B$ is valid.

$\leq$ is clearly transitive and reflexive. For sets of formulas M and N, let $M \leq N$ if for each $A \in M$ there is a $B \in N$ such that $A \leq B$.

▶ **Lemma 19.** *Let π, π' be simple proofs and $\pi \rightsquigarrow \pi'$ by one of the cut reduction steps defined in 2, except contraction. Then $L(G(\pi')) \leq L(G(\pi))$.*

Proof. None of the reduction steps for rule permutations, axioms, or propositional inferences change the grammar[1]. Therefore, the only interesting cases are those of quantifier rules and weakening. Let us consider quantifier inferences first. Let ι be the quantifier inference under discussion. Obviously, there is only a single production for α in $G(\pi)$, namely $\alpha \to t$, and

$$\mathcal{C}_c(\pi) = \alpha \to t \wedge \mathcal{C}'_c(\pi).$$

In $G(\pi')$, α and its single production are deleted and any production $\beta \to s \in P(\pi)$ is replaced by $\beta \to s[\alpha \setminus t]$. Moreover, the constraint formula of $G(\pi')$ is obtained by replacing $\mathcal{C}_c(\pi)$ with $\mathcal{C}'_c(\pi)$ and α with t, respectively, in $\mathcal{C}(\pi)$. Clearly, all other cuts are unaffected by the transformation.

Let $d = \varphi \to \ldots \to C$ be a valid derivation of $G(\pi)$. The derivation d' of $G(\pi')$ that is obtained from d by deleting all applications of $\alpha \to t$ and then simply replacing α with t obviously generates C, so we only need to show that it is valid. There are two cases to

[1] Note, though, that a binary propositional reduction "moves" a conjunction from the constraint formula of one cut to between constraint formulas of two cuts, which makes no semantic difference.

consider here. If α does not occur in d, then $d = d'$ and the validity of d' follows immediately. Now suppose α occurs in d. For every atom $\beta \to s$ in $\mathcal{C}(\pi)$ such that $v_d(\beta \to s) = \top$, clearly $v_{d'}(\beta \to s[\alpha \setminus t]) = \top$. This implies $v_{d'}(\mathcal{C}(\pi')) = \top$ and hence d' is valid in $G(\pi')$. The other direction is proved similarly. Thus, $L(G(\pi')) = L(G(\pi))$.

Now let's consider the case that a cut formula is introduced by weakening on the weak side of c. Let μ be a formula occurrence in the premise on the strong side of c, but not the cut formula, and let $x_1, \ldots, x_n$ be the bound variables in μ. Assume further that μ is an ancestor of a formula occurrence ν in the succedent of the end sequent. If d is any derivation in $G(\pi)$ that begins with $\varphi \to\vdash \widehat{\nu}$, the x_i are eventually replaced with terms t_i in d. Call the end result of this derivation $A(t_1, \ldots, t_n)$. Now consider that in π', μ is introduced via weakening. This means that each x_i has the production $x_i \to w$ in $G(\pi')$. Consequently, we can construct a derivation d' of $A(w, \ldots, w)$ that is valid for $G(\pi')$. Clearly, $A(w, \ldots, w) \leq A(t_1, \ldots, t_n)$.

If the cut formula is introduced by weakening on the strong side, c has no nonterminals and hence contributes nothing to the grammar. In this case, removing the cut clearly changes nothing. ◀

We will need a minor proof transformation that allows us to make some simplifying assumptions later on. The motivation behind this transformation is the following observation: It is never necessary to introduce a strong quantifier twice on the same branch of a proof.

▶ **Definition 20** (Pruning)**.** Let π and π' be proofs of the same end sequent. We say that π' is the result of *"pruning"* π, written as $\pi \rightsquigarrow \pi'$, if π' is obtained from π by the following subproof transformation:

$$
\begin{array}{c}
(\psi) \\
\dfrac{A[x \setminus \beta], \Gamma'' \vdash \Delta''}{\exists x A, \Gamma'' \vdash \Delta''}\ \exists_l \\
\vdots \\
\dfrac{A[x \setminus \alpha], \Gamma' \vdash \Delta'}{\exists x A, \Gamma' \vdash \Delta'}\ \exists_l \\
\vdots \\
\dfrac{C[\exists x A], C[\exists x A], \Gamma \vdash \Delta}{C[\exists x A], \Gamma \vdash \Delta}\ c_l
\end{array}
\quad \rightsquigarrow \quad
\begin{array}{c}
(\psi[\beta \setminus \alpha]) \\
\dfrac{A[x \setminus \alpha], \Gamma'' \vdash \Delta''}{\exists x A, A[x \setminus \alpha], \Gamma'' \vdash \Delta''}\ w_l \\
\vdots \\
\dfrac{\dfrac{A[x \setminus \alpha], A[x \setminus \alpha], \Gamma' \vdash \Delta'}{A[x \setminus \alpha], \Gamma' \vdash \Delta'}\ c_l}{\exists x A, \Gamma' \vdash \Delta'}\ \exists_l \\
\vdots \\
\dfrac{C[\exists x A], C[\exists x A], \Gamma \vdash \Delta}{C[\exists x A], \Gamma \vdash \Delta}\ c_l
\end{array}
$$

We say that a proof is *"pruned"* if it cannot be pruned further.

▶ **Lemma 21.** *Let π, π' be simple proofs such that π' is obtained from π by pruning. Then $L(G(\pi')) \subseteq L(G(\pi))$.*

Proof. We show that every derivation that is valid for $G(\pi')$ can be transformed to one that is valid for $G(\pi)$. This is possible because the eigenvariables that are identified by pruning are associated with the same quantifier and thus have the same productions. Given a derivation d' that is valid for $G(\pi')$, suppose α, β are as in the definition of pruning and d' uses a production $\nu \to t[\beta \setminus \alpha]$. Clearly, $\nu \to t$ is a production of $G(\pi)$ and due to the above considerations, α and β have the same productions. We can therefore replace the step $\nu \to t[\beta \setminus \alpha]$ in d' with $\nu \to t$ and add the required β-productions at any point after that. ◀

For technical reasons, we only allow the reduction of minimal cuts in this lemma. We call a cut minimal if its strong side does not intersect with the weak side of any other cut. It is easy to prove that a minimal cut always exists. The nonterminals of a minimal cut never occur on the right side of productions of other cuts.

▶ **Lemma 22.** *Let π be a pruned simple proof and c a minimal cut in π. If $\pi \rightsquigarrow \pi'$ by reducing c according to a contraction rule, then $L(G(\pi)) = L(G(\pi'))$.*

Proof. We assume that c is Σ_1; the case of a Π_1-cut can be treated by switching the strong and weak sides. Let $G(\pi') = \langle \varphi, N', \rho', \Sigma, P', \mathcal{C}' \rangle$.

First, suppose that the contraction that is reduced is on the left-hand (weak) side of c. The first thing we note is that the only nonterminals that are affected by the proof transformation are those introduced in ψ_2. Due to the minimality of c, there are no quantified cuts in ψ_2 and hence the only eigenvariables therein are those of cuts below c and those of c itself. Let $EV(c) = \{\alpha_1, \dots, \alpha_n\}$. In $G(\pi')$, each α_i is replaced by two new copies α_i' and α_i''. Moreover, if $\tilde{c}$ is a cut in π such that c is on the strong side of $\tilde{c}$, then there might be eigenvariables of $\tilde{c}$ that are introduced within ψ_2. Let $\beta_1, \dots, \beta_m$ be all such eigenvariables; it follows that π' contains two new copies β_i', β_i'' for each of them.

Let us now consider the effects of the reduction on the nonterminals and productions of the end sequent. Let $p : z \to t$ be a production of the end sequent. If t contains no α_i or β_i, p is unchanged; otherwise, p arises from some quantifier inference in ψ_2 that is duplicated along with ψ_2. This means that in $G(\pi')$, p is replaced by two new productions

$$p' : z \to t[\alpha_1 \setminus \alpha_1', \dots, \alpha_n \setminus \alpha_n', \beta_1 \setminus \beta_1', \dots, \beta_m \setminus \beta_m'],$$
$$p'' : z \to t[\alpha_1 \setminus \alpha_1'', \dots, \alpha_n \setminus \alpha_n'', \beta_1 \setminus \beta_1'', \dots, \beta_m \setminus \beta_m''].$$

Now we consider the rest of the grammar. If μ' and μ'' are the two occurrences of A_c on the weak side of c, then one of them is arbitrarily designated as the cut formula of c' and the other as the cut formula of c''; w.l.o.g we assume that μ' is the cut formula of c' and μ'' the cut formula of c''. The productions of α_i are split between α_i' and α_i'' accordingly, that is, if $\alpha_i \to t$ is a production of $G(\pi)$ and t introduces a quantifier in μ', then $\alpha_i' \to t$ is a production of $G(\pi')$ and analogously if t introduces a quantifier in μ''. Note that these cases are not mutually exclusive.

As for the β_i, each of them originates from a cut below c whose weak side is entirely unaffected by the duplication of ψ_2, so β_i' and β_i'' simply inherit the productions of β_i.

Let us now turn to the constraint formula. First of all, the constraint formula of c is necessarily of the form $\mathcal{B}' \vee \mathcal{B}''$; it follows that the constraint formulas of c' and c'' are $\mathcal{B}'$ and $\mathcal{B}''$, respectively, up to replacement of nonterminals by their fresh copies:

$$\mathcal{C}_{c'}(\pi') = \mathcal{B}'\{\alpha_1 \setminus \alpha_1', \dots, \alpha_n \setminus \alpha_n'\},$$
$$\mathcal{C}_{c''}(\pi') = \mathcal{B}''\{\alpha_1 \setminus \alpha_1'', \dots, \alpha_n \setminus \alpha_n''\}$$

Moreover, if ν is any formula occurrence in the conclusion of c originating from ψ_2, then

$$q(\nu, \pi') = q(\nu, \pi)\{\alpha_1 \setminus \alpha_1', \dots, \alpha_n \setminus \alpha_n', \beta_1 \setminus \beta_1', \dots, \beta_m \setminus \beta_m'\} \vee$$
$$\vee\, q(\nu, \pi)\{\alpha_1 \setminus \alpha_1'', \dots, \alpha_n \setminus \alpha_n'', \beta_1 \setminus \beta_1'', \dots, \beta_m \setminus \beta_m''\}$$

because ν is contracted in π'.

If c is above the strong side of $\tilde{c}$, then eigenvariables of $\tilde{c}$ might be duplicated, as noted above. In that case, we obtain the new constraint formula of $\tilde{c}$ by replacing each $\beta_i \to t$ in $\mathcal{C}_{\tilde{c}}(\pi)$ with $\beta_i' \to t \vee \beta_i'' \to t$.

Now let $d = \varphi \to^* s$ be a valid derivation of $G(\pi)$. If no nonterminals belonging to c are used in d then all we have to do to obtain a valid derivation of $G(\pi')$ is replace each

β_i that occurs in d with β'_i. If, on the other hand, such nonterminals are used, then all of them must be produced from nonterminals of the end sequent due to the minimality of c. Let $\alpha_{i_1}, \ldots, \alpha_{i_m}$ be those nonterminals of c that occur in d and assume that each α_{i_j} is later replaced by a term t_j. Then either all of these terms are above μ' or all of them are above μ''. To see this, assume w.l.o.g. that α_{i_1} is later replaced by a term t_1 that introduces a quantifier in μ', but not in μ'' and α_{i_2} by a term t_2 for which the converse is true. Since d is valid, the atom $\alpha_{i_j} \to t_j$ in $\mathcal{C}(\pi)$ is assigned the value $\top$ by v_d and all other atoms beginning with α_{i_j} are assigned $\bot$, due to rigidity. $\mathcal{C}_c(\pi)$ is certainly of the form $\tilde{\mathcal{B}}' \vee \tilde{\mathcal{B}}''$. Since d is valid, either $v_d(\tilde{\mathcal{B}}') = \top$ or $v_d(\tilde{\mathcal{B}}'') = \top$; say the former w.l.o.g. But all α_{i_2}-atoms that occur in $\tilde{\mathcal{B}}'$ evaluate to $\bot$, which is a contradiction.

We now consider the case where all terms produced from the α_{i_j} introduce quantifiers in μ'. In this case, replacing all α_{i_j} in d with α'_{i_j} yields productions of $G(\pi')$. An analogous substitution applied to the c-nonterminals that are introduced by other end sequent nonterminals gives a new derivation d'. The derivation d might also contain some of the β_i. Since the β'_i and the β''_i have the same productions in P' as the β_i do in P, we can simply replace their productions as necessary.

Thus, we obtain a derivation d'' that consists of productions of $G(\pi')$; we now need to show that it is in fact valid. First of all, note that by construction, d'' obeys local rigidity. As for the constraint formula, it is clearly sufficient to show that $v_{d''}$ validates the various conjuncts of $\mathcal{C}(\pi')$.

- If $\tilde{c}$ is a cut with an eigenvariable among the β_i, say β_{i_0}, and β_{i_0} has an associated term t, then the atom $\beta_{i_0} \to t$ in $\mathcal{C}_{\tilde{c}}(\pi)$ is replaced with $\beta'_{i_0} \to t \vee \beta''_{i_0} \to t$ in $\mathcal{C}_{\tilde{c}}(\pi')$ and since $v_d(\mathcal{C}_{\tilde{c}}(\pi)) \leftrightarrow \top$, the same holds for $v_{d''}(\mathcal{C}_{\tilde{c}}(\pi'))$.
- Clearly, $v_{d''}(\mathcal{C}_{c''}(\pi')) = \top$ because none of the α''_i-productions are evaluated by $v_{d''}$.
- $v_{d''}(\mathcal{C}_{c'}(\pi')) = \top$ follows immediately from $v_d(\mathcal{C}_c(\pi)) = \top$.
- The constraint formulas of other cuts and the end sequent are easily seen to be valid under $v_{d''}$.

Conversely, suppose that we have a derivation d' of $G(\pi')$. The first thing we need to establish is that d' can only contain nonterminals of c' or c'', but not both. This is the case because there is no production that contains nonterminals of both and and $\mathcal{C}_{\text{ES}}(\pi')$ forces us to choose either ψ'_2 or ψ''_2 in each derivation. We thus obtain a derivation d of $G(\pi)$ by replacing all $\alpha'_i, \beta'_i, \alpha''_i, \beta''_i$ with their original versions. This d does not violate rigidity due to the considerations above. As in the argument for the other direction, the satisfiability under d of the various parts of $\mathcal{C}$ follows readily from the satisfiability of the corresponding parts of $\mathcal{C}'$.

Now suppose that the contraction happens on the strong side of c. Reducing the contraction leaves us with two new cuts c', c'' whose cut formulas are both A_c. Let μ' and μ'' be the occurrences of A_c that serve as cut formulas for c' and c'' respectively. Each eigenvariable α of c introduces a quantifier in either μ' or μ'' and consequently belongs to either c' or c'' accordingly. Consequently, $EV(c) = EV(c') \dot{\cup} EV(c'')$, where either set on the right-hand side might be empty. Thus, let $EV(c) = \{\alpha_1, \ldots, \alpha_n\}$ and assume for the sake of simplicity that $EV(c') = \{\alpha_1, \ldots, \alpha_k\}$ and $EV(c'') = \{\alpha_{k+1}, \ldots, \alpha_n\}$.

The duplication of the left subproof ψ_1 has extensive effects on the grammar. We will discuss these effects separately for each $\tilde{c} \in \text{QCuts}(\pi)$. First, if $\tilde{c}$ is below c, then c must be on the strong side of $\tilde{c}$ due to c's minimality. As a consequence, it is possible that there are eigenvariables of $\tilde{c}$ that are introduced within ψ_1. If γ is such an eigenvariable, then γ is duplicated, giving rise to eigenvariables γ' and γ''. Each such γ' and γ'' inherits the productions of γ in $G(\pi)$. The constraint formulas of $\tilde{c}$ changes in a straightforward manner, by replacing $\gamma \to t$ with $\gamma' \to t \vee \gamma'' \to t$ for each γ that is duplicated. In the sequel, let $\{\gamma_1, \dots, \gamma_l\}$ be all eigenvariables of the original proof duplicated in this manner.

Next, assume that $\tilde{c}$ is located in ψ_1. In this case, $\tilde{c}$ is replaced with two new cuts $\tilde{c}'$ and $\tilde{c}''$. If $\{\beta_1, \dots, \beta_m\}$ are all eigenvariables that belong to such cuts, then clearly each of them is replaced by two new copies β_i' and β_i''. The productions of these duplicates work out to

$$P'_{\beta'} = P_{\beta_i}\{\bar{\beta} \setminus \bar{\beta}', \bar{\gamma} \setminus \bar{\gamma}'\},$$
$$P'_{\beta''} = P_{\beta_i}\{\bar{\beta} \setminus \bar{\beta}'', \bar{\gamma} \setminus \bar{\gamma}''\}$$

for each $i \in \{1, \dots, m\}$. Similarly, $\tilde{c}'$ and $\tilde{c}''$ have the constraint formulas

$$\mathcal{C}_{\tilde{c}'} = \mathcal{C}_{\tilde{c}}\{\beta_1 \setminus \beta_1', \dots, \beta_m \setminus \beta_m'\},$$
$$\mathcal{C}_{\tilde{c}''} = \mathcal{C}_{\tilde{c}}\{\beta_1 \setminus \beta_1'', \dots, \beta_m \setminus \beta_m''\}$$

respectively. The final case to consider is that of c itself: The productions of the α_i in $G(\pi')$ work out to

$$P'_{\alpha_i} = P_{\alpha_i}[\beta_1 \setminus \beta_1', \dots, \beta_m \setminus \beta_m', \gamma_1 \setminus \gamma_1', \dots, \gamma_k \setminus \gamma_l'] \text{ for } i \leq k,$$
$$P'_{\alpha_i} = P_{\alpha_i}[\beta_1 \setminus \beta_1'', \dots, \beta_m \setminus \beta_m'', \gamma_1 \setminus \gamma_1'', \dots, \gamma_k \setminus \gamma_l''] \text{ for } i > k.$$

The constraint formula of c' can be obtained from $\mathcal{C}_c$ by replacing each literal $\alpha_i \to t$ that occurs in it with $\alpha_i \to t[\beta_1 \setminus \beta_1', \dots, \beta_m \setminus \beta_m', \gamma_1 \setminus \gamma_1', \dots, \gamma_k \setminus \gamma_k']$ (for $i \leq k$) or removing it (for $i > k$). An analogous transformation yields $\mathcal{C}_{c''}$. If $\tilde{c}$ is any other quantified cut, then $\tilde{c}$ is either within the strong side of c or on a different branch of the proof from c. The first case is impossible due to minimality of c and in the second case, $\tilde{c}$ is unaffected by the proof transformation.

The last thing that needs to be taken care of are the productions and constraint formula of the end sequent. Each production $z_i \to t$ is replaced by

$$z_i \to t[\beta_1 \setminus \beta_1', \dots, \beta_m \setminus \beta_m', \gamma_1 \setminus \gamma_1', \dots, \gamma_k \setminus \gamma_k'] \text{ and}$$
$$z_i \to t[\beta_1 \setminus \beta_1'', \dots, \beta_m \setminus \beta_m'', \gamma_1 \setminus \gamma_1'', \dots, \gamma_k \setminus \gamma_k''].$$

If t does not contain any β_i or γ_i, then both of these duplicates obviously coincide with the original production and it simply carries over to $G(\pi')$. As for $\mathcal{C}_{\text{ES}}(\pi')$, there are formulas $\mathcal{B}_1, \dots, \mathcal{B}_r$ such that

$$\mathcal{C}_{\text{ES}}(\pi) = \mathcal{C}[\mathcal{B}_1, \dots \mathcal{B}_r] \text{ and}$$
$$\begin{aligned}\mathcal{C}_{\text{ES}}(\pi') = \mathcal{C}[&\mathcal{B}_1[\beta_1 \setminus \beta_1', \dots, \beta_m \setminus \beta_m', \gamma_1 \setminus \gamma_1', \dots, \gamma_k \setminus \gamma_k'] \vee \\ &\vee \mathcal{B}_1[\beta_1 \setminus \beta_1'', \dots, \beta_m \setminus \beta_m'', \gamma_1 \setminus \gamma_1'', \dots, \gamma_k \setminus \gamma_k''], \\ &\dots \\ &\quad \mathcal{B}_r[\beta_1 \setminus \beta_1', \dots, \beta_m \setminus \beta_m', \gamma_1 \setminus \gamma_1', \dots, \gamma_k \setminus \gamma_k'] \vee \\ &\vee \mathcal{B}_r[\beta_1 \setminus \beta_1, \dots, \beta_m \setminus \beta_m'', \gamma_1 \setminus \gamma_1'', \dots, \gamma_k \setminus \gamma_k'']].\end{aligned}$$

Let d be a valid derivation of $G(\pi)$. If nonterminals of c occur in d, then due to the minimality of c they can only be introduced from nonterminals of the end sequent. Let $\alpha_{i_1}, \dots, \alpha_{i_r}$ be those nonterminals of c that are used in d and are later replaced by terms $t_1, \dots, t_r$. For each i_j, we replace the production $\alpha_{i_j} \to t_j$ with $\alpha_{i_j} \to t_j[\bar{\beta} \setminus \bar{\beta}', \bar{\gamma} \setminus \bar{\gamma}']$ if $i_j \leq k$ or $\alpha_{i_j} \to t_j[\bar{\beta} \setminus \bar{\beta}'', \bar{\gamma} \setminus \bar{\gamma}'']$ if $i_j > k$. Also, if $z_i \to t$ is a production of the end sequent in d, we replace it with $z_i \to t[\bar{\beta} \setminus \bar{\beta}', \bar{\gamma} \setminus \bar{\gamma}']$, obtaining a new derivation d'. This can lead to

d' containing both β_i' and β_i'' for some i, and similarly for the γ_i. Due to total rigidity, d uses at most one production for each β_i and γ_i and we can simply replace any such production by one or both of its two variants in the new grammar, according to whether one or both copies of the respective nonterminal occur in d'. We call the derivation obtained by this process d''.

As before, it is sufficient to show that d'' is totally rigid and validates the conjuncts of $\mathcal{C}(\pi')$. $v_{d''}(\mathcal{C}_{c'}) = \top$ because up to renaming, the literals of $\mathcal{C}_{c'}$ are a subset of those of $\mathcal{C}_c$ and $v_d(\mathcal{C}_c) = \top$. The satisfiability of $v_{d''}(\mathcal{C}_{c''})$ is shown in an analogous manner. The constraint formulas of all other cuts are similarly easy to deal with because they contain the same substitutions relative to their original counterparts as d'' does to d. $v_{d''}(\mathcal{C}_{\mathrm{ES}}(\pi')) = \top$ is immediately obvious.

Now suppose that we have a valid derivation d' of $G(\pi')$. First of all, there are some important conclusions to be drawn from the form of $\mathcal{C}_{\mathrm{ES}}(\pi')$: Let x, y be nonterminals of the end sequent such that x dominates y. If some production $x \to t(\bar{\alpha})$ is used in d', no production of y that is used in d' can contain any of the β_i' or γ_i' (or their $''$-versions), and vice versa. Moreover, if there is a production $x \to t_i(\bar{\beta}', \bar{\gamma}')$ in d', then productions $y \to t_j(\bar{\beta}'', \bar{\gamma}'')$ cannot occur in d', and analogously with the $'$- and $''$-nonterminals changed around. Since π is pruned, no term in ψ_2 contains two eigenvariables that introduce the same quantifier. These facts imply that we can simply replace all $'$- and $''$-nonterminals by their original versions without violating total rigidity. The argument that the resulting derivation d is valid then goes through just as in the previous cases. ◀

We can now finally prove the main result of this paper:

▶ **Theorem 16.** *Let π be a simple proof of $\Gamma \vdash \Delta$. Then $L(G(\pi))$ is an Herbrand set of $\Gamma \vdash \Delta$.*

Proof. By combining Lemmas 17, 19, 21, and 22. ◀

6 Conclusion

In this paper we have given a description of the Herbrand set induced by a proof with non-prenex Π_1 and Σ_1 cuts in terms of a tree grammar. This is a considerable extension of the previously existing work for prenex formulas [8] since the structure of sequent calculus proofs and the dynamics of cut-elimination changes significantly when non-prenex cuts are allowed. The central tool for this description are constraint grammars, which permit capturing the dependencies of the quantifier instantiations in the proof.

Applications of the connection between formal language theory and proof theory described in [8] for prenex Π_1 and Σ_1 cuts include results on Herbrand-confluence [12], cut-introduction [11, 10, 9], inductive theorem proving [4], and proof complexity [3]. In addition, this connection has recently been extended to prenex Π_2 and Σ_2 cuts [1]. In this line of research, this paper is the first to consider *non-prenex* formulas and thus opens the way for extending the above results and techniques to non-prenex cuts and induction formulas.

References

1 Bahareh Afshari, Stefan Hetzl, and Graham E. Leigh. Herbrand disjunctions, cut elimination and context-free tree grammars. to appear at Typed Lambda Calculi and Applications (TLCA) 2015.

2 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.

3 Sebastian Eberhard and Stefan Hetzl. Compressibility of finite languages by grammars. to appear at Descriptional Complexity of Formal Systems (DCFS) 2015, preprint available at `http://www.logic.at/people/hetzl/research/`.

4 Sebastian Eberhard and Stefan Hetzl. Inductive theorem proving based on tree grammars. *Annals of Pure and Applied Logic*, 166(6):665–700, 2015.

5 Philipp Gerhardy and Ulrich Kohlenbach. Extracting Herbrand Disjunctions by Functional Interpretation. *Archive for Mathematical Logic*, 44:633–644, 2005.

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

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

8 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*, pages 301–312. Springer, 2012.

9 Stefan Hetzl, Alexander Leitsch, Giselle Reis, Janos Tapolczai, and Daniel Weller. Introducing Quantified Cuts in Logic with Equality. In Stéphane Demri, Deepak Kapur, and Christoph Weidenbach, editors, *Automated Reasoning – 7th International Joint Conference, IJCAR*, volume 8562 of *Lecture Notes in Computer Science*, pages 240–254. Springer, 2014.

10 Stefan Hetzl, Alexander Leitsch, Giselle Reis, and Daniel Weller. Algorithmic introduction of quantified cuts. *Theoretical Computer Science*, 549:1–16, 2014.

11 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.

12 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.

13 Stefan Hetzl and Daniel Weller. Expansion trees with cut. preprint, available at `http://arxiv.org/abs/1308.0428`, 2013.

14 David Hilbert and Paul Bernays. *Grundlagen der Mathematik II.* Springer, 1939.

15 Markus Lohrey. Algorithmics on SLP-compressed strings: A survey. *Groups Complexity Cryptology*, 4(2):241–299, 2012.

16 Richard McKinley. Proof nets for Herbrand's Theorem. *ACM Transactions on Computational Logic*, 14(1):5:1–5:31, 2013.

17 Dale A. Miller. A compact representation of proofs. *Studia Logica*, 46(4):347–370, 1987.

18 V.P. Orevkov. Lower bounds for increasing complexity of derivations after cut elimination. *Zapiski Nauchnykh Seminarov Leningradskogo Otdeleniya Matematicheskogo Instituta*, 88:137–161, 1979.

19 Pavel Pudlák. The Lengths of Proofs. In Sam Buss, editor, *Handbook of Proof Theory*, pages 547–637. Elsevier, 1998.

20 Richard Statman. Lower bounds on Herbrand's theorem. *Proceedings of the American Mathematical Society*, 75:104–107, 1979.

21 Sebastian Zivota. Cuts Without Quantifier Alternations and Their Effect on Expansion Trees. Master's thesis, Vienna University of Technology, 2014.