FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Herbrand Sequente und die skolemisierungs-freie CERES Methode

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieurin

im Rahmen des Studiums

Computational Intelligence

eingereicht von

Anela Lolic
Matrikelnummer 1055207

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Univ.Prof. Dr. phil. Alexander Leitsch

Wien, 16. August 2015

Anela Lolic   Alexander Leitsch

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.ac.at

FAKULTÄT FÜR !NFORMATIK

Faculty of Informatics

Herbrand sequents and the Skolem-free CERES method

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieurin

in

Computational Intelligence

by

Anela Lolic

Registration Number 1055207

to the Faculty of Informatics

at the Vienna University of Technology

Advisor: Univ.Prof. Dr. phil. Alexander Leitsch

Vienna, 16th August, 2015

Anela Lolic | Alexander Leitsch

Erklärung zur Verfassung der Arbeit

Anela Lolic
Kandlgasse 38/4, 1070 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 16. August 2015

Anela Lolic

Acknowledgements

I would like to thank my advisor Professor Alexander Leitsch. First of all he provided me a very interesting topic and the ability to work in the area of proof theory. The time I spent on working on this thesis was the most interesting time during my studies and I am very grateful that I had the opportunity to work in such an interesting field of logic. I am also thankful to my advisor for spending so much time in discussing my work and for always having time for all of my issues.

I would also like to thank my family and especially my parents, for supporting me and for so many other things. Without them, I never would have been able to achieve a goal like this.

Kurzfassung

Ein Teilgebiet der mathematischen Logik ist die Beweistheorie, welche Beweise als formale Objekte betrachtet und deren Eigenschaften untersucht. Einer der wichtigsten Sätze der Beweistheorie ist der Satz der Schnittelimination. Dieser wurde von Gerhard Gentzen bewiesen und besagt, dass die sogenannte Schnittregel aus einem formalen Beweissystem (in der Art des Sequentialkalküls **LK**) immer enfernt werden kann. Die Schnittelimination kann in konkreten mathematischen Beweisen als eine Methode zum Entfernen von Hilfssätzen (Lemmata) angesehen werden.

Eine Eigenschaft schnittfreier Beweise ist, dass sie nur Teilformeln der Formeln im zu beweisenden Satz enthalten, d.h. dass sie die Teilformel-Eigenschaft besitzen.

Gentzen's Methode zur Schnittelimination wird als reduktive Schnittelimination betrachtet. Hier betrachtet man nicht den gesamten Beweis, sondern führt lokale Beweistransformationen an einem Teil des gesamten Beweises durch.

CERES (cut-elimination by resolution) stellt einen alternativen Ansatz dar. Hier werden alle Schnitte gleichzeitig analysiert und somit wird die globale Struktur des Beweises berücksichtigt. Grob gesagt wird eine widerlegbare Klauselmenge extrahiert, welche die Struktur eines Beweises mit Schnitten repräsentiert. Die Resolutionswiderlegung dieser Klauselmenge dient als Skelett für einen Beweis, der höchstens atomare Schnitte enthält.

CERES$^\omega$ wurde als CERES-Methode für Logik höherer Ordnung entwickelt und arbeitet, im Gegensatz zu CERES, mit skolemfreien End-Sequenten. Für die Schnittelimination wird ein neuer Sequentialkalkül, $\mathbf{LK}_{sk}$, eingeführt, der keine Schnittregel enthält.

Im Zuge dieser Arbeit wurde die Idee von skolemfreien Beweisen auf Logik erster Ordnung überführt, um eine CERES-Methode zu entwickeln, welche auch mit starken Quantoren im End-Sequent arbeiten kann. Wir konzentrieren uns auf die Extraktion von Herbrand Sequenten und nicht auf das Erzeugen der ACNF, welche ein Beweis mit höchstens atomaren Schnitten ist. Herbrand Sequente wurden in der ursprünglichen CERES-Methode aus der ACNF erzeugt. Wir zeigen, dass man für die Extraktion die ACNF nicht benötigt und dass man die Herbrand Sequente bereits aus der Resolutionswiderlegung und den zugehörigen Projektionen gewinnen kann. Unsere Methode zur Extraktion von Herbrand Sequenten ist exponentiell schneller als die ursprüngliche skolemfreie Methode.

Abstract

A branch of mathematical logic is proof theory, which considers proofs as formal objects and is concerned with the analysis of their properties. One of the main theorems in proof theory is the cut-elimination theorem. It was proved by Gerhard Gentzen and states that the so-called cut-rule can always be eliminated from a formal proof system in the style of the sequent calculus **LK**. In real mathematical proofs, cut-elimination can be regarded as a method for eliminating lemmas.

One property of cut-free proofs is that they only use subformulas of the formulas that are already present in the statement which has to be proved, i.e. they have the subformula-property.

Gentzen's cut-elimination method is regarded as reductive cut-elimination. This method does not analyse the whole proof, but performs local proof rewriting steps on small parts of the proof.

Another approach to cut-elimination is the method CERES (cut-elimination by resolution). In this method all cuts are analysed simultaneously and hence the global structure of the proof is taken into account. Roughly speaking, CERES extracts an unsatisfiable set of clauses, that encodes the structure of a proof containing cuts. A resolution refutation of this set of clauses serves as a skeleton for a proof containing at most atomic cuts.

CERES$^{\omega}$ was developed as CERES-method for higher-order logic and works, in contrast to CERES, with Skolem-free end-sequents. For the cut-elimination a new sequent calculus, $\mathbf{LK}_{sk}$, is introduced which does not contain the cut-rule.

In the course of this work the idea of Skolem-free proofs was transferred to first-order logic, to gain a CERES-method which works in the presence of strong quantifiers in the end-sequent. We concentrate on the extraction of Herbrand sequents instead of the construction of the ACNF, a proof with at most atomic cuts. In the original CERES-method Herbrand sequents were extracted from the ACNF. We show, that the ACNF is not needed for the extraction and that the Herbrand sequents can be extracted from the resolution refutation and the corresponding projections. Our method for the extraction of Herbrand sequents is exponentially faster than the original, Skolem-free method.

Contents

CHAPTER 1

Introduction

One of the main objectives in mathematics is to show that mathematical statements are valid. The subfield of mathematical logic, proof theory, is devoted to the study of proofs as mathematical objects. Since proofs can be regarded as finite objects, i.e. finite strings of symbols, in the beginning of proof theory it was believed that showing the consistency of logical theories using purely finitistic methods could be achieved by using proof-theoretic methods. However, Gödel proved in his seminal papers that it is not possible to prove the consistency of mathematical theories using purely finitistic means.

Gentzen invented the sequent calculi **LK** and **LJ** for classical and intuitionistic logic, respectively, in [9] which have a great variety of use. His calculi contain the cut-rule, which allows the use of lemmas (i.e. intermediary statements) in proofs. The main result of the paper was the cut-elimination theorem, which basically states that any theorem of first-order logic can be proved without detours, i.e. without the use of instances of the cut rule [9]. Cut-free proofs have the subformula property, which means that all formulas used in the proof are (instances of) subformulas of the statement to be proved [5], [15]. What follows directly from the subformula property is the consistency of both, **LK** and **LJ**. Indeed, if there was a proof of the empty sequent, it would be provable without using the cut-rule and this is impossible by the subformula property [20].

Cut-elimination can also be applied to real mathematical proofs. One example is Girard's analysis (see in [11]) of Fürstenberg and Weiss' topological proof [8] of Van der Waerden's theorem [21] on partitions. After cut-elimination was applied on the proof of Fürstenberg and Weiss, the result was Van der Waerden's original elementary proof [15].

There exist different methods to prove the cut-elimination theorem. The original idea by Gentzen can be regarded as a rewrite system on proofs that is applied according to a specific strategy. It is called reductive cut-elimination.

Baaz and Leitsch introduced an alternative cut-elimination method based on resolution called CERES (cut elimination by resolution) [3]. The technique is novel since it relies on the resolution method from automated-theorem proving. The method CERES takes the global structure of an **LK**-proof into account, in contrast to reductive cut-elimination which operates on small parts of the proof. The general procedure of CERES can be described as follows. First extract the characteristic clause set (an unsatisfiable set of clauses encoding the structure of a proof that contains cuts) from the input proof. Then obtain a resolution refutation γ of the characteristic clause set, which serves as a skeleton for a proof ϕ containing at most atomic cuts. Finally transform the resolution refutation γ into ϕ by replacing its leaves by so-called projections (i.e. cut-free parts of the original proof) [6].

The CERES method was originally defined as a cut-elimination method for first-order logic. For higher-order logic a different method, CERES$^\omega$, was defined [13], [22]. In first-order logic, the CERES method is restricted to work on Skolemized proofs, i.e. proofs of theorems which do not contain strong quantifiers. Skolemized proofs have the property that they do not contain strong quantifier inferences operating on end-sequent ancestors. The strong quantifier rules are the only rules in **LK** that impose restrictions on the context of the rule. Without these restrictions, proof transformations can be performed in a more flexible way. In higher-order logic, the usual notion of a Skolemized proof has not the same consequence as in first-order logic, hence eigenvariable conditions may be violated. Therefore the method CERES$^\omega$ was defined, which works on a new cut-free sequent calculus $\mathbf{LK}_{sk}$ which introduces quantifiers from Skolem terms. It is shown in [13] and [22] that $\mathbf{LK}_{sk}$ is sound and can be translated into **LK**, yielding a cut-elimination method for proofs in higher-order logic.

We use the idea of CERES$^\omega$ to define a new CERES-method which works with proofs that are not Skolemized, i.e. proofs of theorems which contain strong quantifiers. Therefore we will also use the sequent calculus $\mathbf{LK}_{sk}$.

We will focus on Herbrand sequent extraction instead on generating the ACNF (a proof with at most atomic cuts). In the ordinary CERES-method, the Herbrand sequents can be extracted out of the ACNF. Since in the Skolem-free CERES-method, the generation of the ACNF from projections and the resolution refutation is very complicated and expensive, we will skip this transformation and extract the Herbrand sequents from the projections and the resolution refutation.

We will also show that the extraction of Herbrand sequents can be sped-up exponentially by omitting some transformations (i.e. by omitting the transformation of an $\mathbf{LK}_{sk}$-proof into an **LK**-proof, this transformation was needed to show the soundness of $\mathbf{LK}_{sk}$). Hence, our method for CERES is Skolem-free and Herbrand sequents are extracted directly from $\mathbf{LK}_{sk}$-proofs.

To sum up, the novel contribution of this thesis consists in the development of a more efficient CERES$^\omega$-method for first-order logic. As the key information in a proof (of a prenex end-sequent) is stored in its Herbrand sequent, producing an ACNF first (and then extracting the Herbrand sequent) is actually a detour; this detour can be avoided in the CERES-method (using the resolution refutation an projections only) and leads to an exponential improvement.

The result is an efficient Skolemization-free CERES-method for first-order logic.

1.1 Structure of the Thesis

To fix notation and terminology we present the basic notions and definitions in Chapter 2.

In Chapter 3 a general overview on cut-elimination and its important consequences is given. We conclude this chapter with the definition of a proof rewriting system for cut-elimination based on Gentzen's proof of the cut-elimination theorem.

In Chapter 4 we will introduce CERES (cut-elimination by resolution) and prove some of its most important properties.

The definition of the CERES-method for higher-order logic (CERES$^\omega$) will follow in Chapter 5. We will state and prove some of the most important properties.

In Chapter 6 we will give a brief complexity analysis of the method CERES.

Chapter 7 is devoted to the proof of the main result of this thesis, namely that Herbrand sequents can be constructed from the resolution refutation and the corresponding projections and that our method for the extraction of Herbrand sequents outperforms the old one. To show this, we will introduce a Skolem-free CERES-method for first-order logic and prove that the Herbrand sequents can be extracted from the resolution refutation and the corresponding projections, instead of the ACNF. Then we will show that we can speed-up the extraction of Herbrand sequents by omitting some superfluous proof-transformations.

This thesis is then concluded in Chapter 8 where we summarize the main results.

CHAPTER 2

Preliminaries

To fix notion and terminology the following chapter will give a short overview on the logical notions used in this thesis. First of all, in Section 2.1 the syntax and semantics of classical first-order logic will be introduced. Then, in Section 2.2 the sequent calculus **LK** will be defined and finally, in Section 2.3 we will introduce the resolution calculus.

2.1 First-Order Logic

Syntax
The following definitions are based on and taken from [14].

Definition 2.1.1. Language. The language $\mathscr{L}$ of classical first-order logic consists of the following elements:

- a countably infinite set of individual variables V,
- a countably infinite set of constant symbols CS,
- a countably infinite set of function symbols $FS = \bigcup_{i=1}^{\infty} FS_i$, where all sets FS_i, for $i > 1$, are countably infinite (FS_i is the set of $i-$ary function symbols),
- a countably infinite set of predicate symbols $PS = \bigcup_{i=1}^{\infty} PS_i$, where all sets PS_i, for $i > 1$, are countably infinite (PS_i is the set of $i-$ary predicate symbols),
- the logical connectives $\wedge$, $\vee$, $\neg$ and $\rightarrow$,
- the quantifiers $\forall$ and $\exists$,
- $\top$ (verum) and $\bot$ (falsum).

Unless stated otherwise, we will use the following notational conventions [14]:

- Variables: $x, y, z, u, v, w, x_1, y_1, ...$
- Constant symbols: $a, b, c, d, e, a_1, b_1, ...$
- Function symbols: $f, g, h, f_1, g_1, ...$
- Predicate symbols: $P, Q, R, P_1, Q_1, ...$

Definition 2.1.2. Term. The set of terms T is inductively defined as follows:

- $V \subseteq T$ (variables are terms),
- $CS \subseteq T$ (constant symbols are terms),
- If $t_1, ..., t_n \in T$ and $f \in FS_n$ with $n > 1$, then $f(t_1, ...t_n) \in T$.
- No other objects are terms.

Statements like "No other objects are..." will henceforth be omitted and will be considered as included in the concept of "definition" [14].

If $t = f(t_1, ..., t_n)$, for an $f \in FS_n$ and terms $t_1, ..., t_n$, then t is called a functional term. The terms t_i are called the arguments of t. Variables and constant symbols have no arguments.

The occurrence of terms can be defined inductively: A term s occurs in a term t if either $s = t$ holds or s occurs in an argument of t.

The set of all variables occurring in a term t is denoted by $V(t)$. A term t with $V(t) = \emptyset$ is called a ground term.

Definition 2.1.3. Formula. The set of first-order logic formulas PL is inductively defined as follows:

1. If $P \in PS_n$, where $n > 1$ and $t_1, ..., t_n \in T$, then $P(t_1, ..., t_n) \in PL$,
2. $\top \in PL$ and $\bot \in PL$,
3. If $A \in PL$, then also $\neg A \in PL$,
4. If $A, B \in PL$, then also $A \wedge B \in PL$, $A \vee B \in PL$ and $A \rightarrow B \in PL$,
5. If $A \in PL$ and $x \in V$, then $(\forall x)A \in PL$,
6. If $A \in PL$ and $x \in V$, then $(\exists x)A \in PL$.

Formulas obtained by Definition 2.1.3, 1. are called atomic formulas or atoms, and the t_1,...,t_n are called the arguments of $P(t_1,...,t_n)$. Let A be a formula such that $A = A_1 \odot A_2$, $A = \neg B$ or $A = (Qx)B$ for $\odot \in \{\wedge, \vee, \rightarrow\}$ and $Q \in \{\forall, \exists\}$. Then A_1, A_2 and B are called immediate subformulas of A.

A formula A occurs in a formula B if either $A = B$ or A occurs in an immediate subformula of B. A is called a subformula of B if A occurs in B. In 5. and 6. A

and all terms occurring in A and all its subformulas are in the scope of $(\forall x)$ and $(\exists x)$, respectively.

Let s be a term and A be an atomic formula. If s occurs in an argument of A, then s occurs in A. If A is an arbitrary formula, then s occurs in A if it occurs in some subformula of A.

Definition 2.1.4. Free and Bounded Occurrences of Variables. Let A be an atomic formula and x be a variable occurring in A, then x occurs free in A. If x occurs free in A and B is of the form $A \odot C$, $C \odot A$, $\neg A$ or $(Qy)A$ for $\odot \in \{\land, \lor, \to\}$, $y \neq x$, $Q \in \{\forall, \exists\}$, then x occurs free in B.

x occurs bounded in A if there is a subformula of A of the form $(Qx)B$ such that x occurs free in B.

A closed formula or a sentence is a formula where all variables are bounded. If a formula does not contain bounded variables, it is called open.

Definition 2.1.5. Universal Closure. If A is an open formula containing the free variables $x_1, ..., x_n$, then $(\forall x_1), ..., (\forall x_n)A$ is called the universal closure of A.

The universal closure of a formula A is not unique, since the order of the variables is not fixed. But all closures are semantically equivalent [14].

The following definition is based on [6] Definition 3.1.2.

Definition 2.1.6. Position. We inductively define positions within terms as follows:

1. If $t \in V$ or $t \in CS$ then ϵ is a position in t and $t.\epsilon = t$.

2. If $t = f(t_1, ..., t_n)$ where $f \in FS_n$ and $t_1, ..., t_n \in T$, then ϵ is a position in t and $t.\epsilon = t$. Let μ be a position in a t_j for $1 \leq j \leq n$, $\mu = (k_1, ..., k_l)$ and $t_j.\mu = s$, then ν for $\nu = (j, k_1, ..., k_l)$ is a position in t and $t.\nu = s$.

The following definitions are taken from [6].

Positions are used to locate subterms in a term and to perform replacements on subterms. A subterm s of t is just a term with $t.\nu = s$ for some position ν in t. Let $t.\nu = s$, then $t[r]_\nu$ is the term t after replacement of s on position ν by r, in particular $t[r]_\nu.\nu = r$. Let P be a set of position on t, then $t[r]_P$ is defined from t by replacing all $t.\nu$ with $\nu \in P$ by r.

The following example illustrates the definition of a position and is taken from [6] Example 3.1.2.

Example 2.1.1. Let $t = f(f(\alpha, \beta), \alpha)$ be a term. Then

$$t.\epsilon = t$$

$$t.(1) = f(\alpha, \beta)$$

$$t.(2) = \alpha$$

$$t.(1,1) = \alpha$$
$$t.(1,2) = \beta$$
$$t[g(a)].(1,1) = f(f(g(a),\beta),\alpha)$$

Positions within formulas can be defined in the same way, just consider all formulas as terms.

The following definition is taken from [6] Definition 3.1.3.

Definition 2.1.7. Substitution. A substitution is a mapping from $V_f \cup V_b$ to the set of terms such that $\sigma(v) \neq v$ for only finitely many $v \in V_f \cup V_b$. If σ is a substitution with $\sigma(x_i) = t_i$ for $x_i \neq t_i$ $(1 \leq i \leq n)$ and $\sigma(v) = v$ for $v \notin \{x_1,...,x_n\}$ then we denote σ by $\{x_1 \leftarrow t_1,...,x_n \leftarrow t_n\}$. We call the set $\{x_1,...,x_n\}$ the domain of σ and denote it by $dom(\sigma)$. Substitutions are written in postfix, i.e. we write $F\sigma$ instead of $\sigma(F)$.

Substitutions can be extended to terms, atoms and formulas in a similar way.

Definition 2.1.8. [6] Definition 3.1.4 A substitution σ is called more general than a substitution η $(\sigma \leq_s \eta)$ if there exists a substitution μ such that $\eta = \sigma\mu$.

The next definition is based on [6] Definition 3.1.6.

Definition 2.1.9. Logical complexity of formulas. If F is a formula in PL then the complexity $comp(F)$ is the number of logical symbols occurring in F. We define

- If F is an atomic formula then $comp(F) = 0$,
- If $F = A \odot B$ for $\odot \in \{\wedge,\vee,\rightarrow\}$ then $comp(F) = 1 + comp(A) + comp(B)$,
- If $F = \neg A$ or $F = (Qx)A$ for $Q \in \{\forall,\exists\}$ and $x \in V$ then $comp(F) = 1 + comp(A)$.

Semantics

The following definition is based on [14] Definition 2.1.6.

Definition 2.1.10. Interpretation. An interpretation of a formula $F \in PL$ is a triple $\mathscr{M} = (D,\Phi,I)$ with the following properties

1. the domain D of $\mathscr{M}$ is a nonempty set.
2. Φ is a mapping defined on $CS(F) \cup FS(F) \cup PS(F)$ such that
   a) $\Phi(c) \in D$ for $c \in CS(F)$,
   b) $\Phi(f) : D^n \rightarrow D$ for $f \in FS_n(F)$,
   c) $\Phi(P) \subseteq D^n$ for $P \in PS_n(F)$
3. the environment or variable assignment $I : V \rightarrow D$.

Interpretations are the basis for the interpretation functions $u_{\mathscr{M}}$ for terms and $v_{\mathscr{M}}$ for formulas. The next definition is based on [14].

Definition 2.1.11. Interpretation function. Let $F \in PL$ and $\mathscr{M}$ be an interpretation of F, we define the interpretation function $u_{\mathscr{M}} : T(F) \to D$

1. If $x \in V$ then $u_{\mathscr{M}}(x) = I(x)$,
2. if $c \in CS(F)$ then $u_{\mathscr{M}}(c) = \Phi(c)$,
3. if $f(t_1, ..., t_n) \in T(F)$ then $u_{\mathscr{M}}(f(t_1, ..., t_n)) = \Phi(f)(u_{\mathscr{M}}(t_1), ..., u_{\mathscr{M}}(t_n))$,

where $T(F)$ denotes the set of terms occurring in F.

The next definition is based on [14] Definition 2.1.17.

Definition 2.1.12. Equivalence of interpretations. Let $\mathscr{M}$ and $\mathscr{M}'$ be two interpretations of a formula F. $\mathscr{M}$ and $\mathscr{M}'$ are called equivalent modulo $x_1, ..., x_n$ if there are D, ϕ, I, J such that $\mathscr{M} = (D, \Phi, I)$, $\mathscr{M}' = (D, \Phi, J)$ and $I(v) = J(v)$ for $v \in V \backslash \{x_1, ..., x_n\}$. If $\mathscr{M}$ and $\mathscr{M}'$ are equivalent modulo x, we write $\mathscr{M} \sim_x \mathscr{M}'$.

Now we are able to define the evaluation of formulas in $PL(F)$ via an interpretation $\mathscr{M}$, where $PL(F)$ is the set of formulas over the language of F [14]. The following definition is based on [14].

Definition 2.1.13. Evaluation of formulas. Let $F \in PL$ and $\mathscr{M} = (D, \Phi, I)$ be an interpretation of F. Then $v_{\mathscr{M}} : PL(F) \to \{true, false\}$ is defined inductively over the structure of formulas in $PL(F)$.

1. If A is an atomic formula in $PL(F)$ and $A = P(t_1, ..., t_n)$ then $v_{\mathscr{M}}(A) = true$ if and only if $(u_{\mathscr{M}}(t_1), ..., u_{\mathscr{M}}(t_n)) \in \Phi(P)$,
2. $v_{\mathscr{M}}(\top) = true$ and $v_{\mathscr{M}}(\bot) = false$,
3. $v_{\mathscr{M}}(\neg A) = true$ if and only if $v_{\mathscr{M}}(A) = false$,
4. $v_{\mathscr{M}}(A \wedge B) = true$ if and only if $v_{\mathscr{M}}(A) = true$ and $v_{\mathscr{M}}(B) = true$,
5. $v_{\mathscr{M}}(A \vee B) = true$ if and only if $v_{\mathscr{M}}(A) = true$ or $v_{\mathscr{M}}(B) = true$,
6. $v_{\mathscr{M}}(A \to B) = true$ if and only if $v_{\mathscr{M}}(A) = false$ or $v_{\mathscr{M}}(B) = true$,
7. $v_{\mathscr{M}}((\forall x)A) = true$ if and only if for all $\mathscr{M}'$ such that $\mathscr{M} \sim_x \mathscr{M}'$ we have $v_{\mathscr{M}'}(A) = true$,
8. $v_{\mathscr{M}}((\exists x)A) = true$ if and only if for some $\mathscr{M}'$ such that $\mathscr{M} \sim_x \mathscr{M}'$ we have $v_{\mathscr{M}'}(A) = true$,

where $A, B \in PL(F)$.

An interpretation $\mathscr{M}$ of A verifies A if $v_{\mathscr{M}}(A) = true$, if $v_{\mathscr{M}}(A) = false$ we say that $\mathscr{M}$ falsifies A [14].
The following definition is based on [14] Definition 2.1.8.

Definition 2.1.14. Model. Let A be a formula containing the free variables $x_1, ..., x_n$ and $\mathscr{M}$ be an interpretation of A. $\mathscr{M}$ is a model of A if all $\mathscr{M}'$ such that $\mathscr{M} \sim_{x_1,...,x_n} \mathscr{M}'$ verify A. If A is a closed formula $\mathscr{M}$ is a model of A if and only if $\mathscr{M}$ verifies A. We denote that $\mathscr{M}$ is a model of A by $\mathscr{M} \models A$.

The next definition is an extension of [14] Definition 2.1.9 that contains the definition of unsatisfiability.

Definition 2.1.15. (Un)satisfiability and validity. Let $F, G \in PL$ be arbitrary. Then

- F is called satisfiable if F has a model.
- F is called unsatisfiable if F is not satisfiable.
- F is called valid if every interpretation of F is a model of F.
- F and G are logically equivalent ($F \equiv G$) if F and G have exactly the same models.
- F and G are satisfiability-equivalent if F is satisfiable if and only if G is satisfiable ($F \equiv_{sat} G$).

Having defined the syntax and semantics of classical first-order logic, we are able to introduce a formal proof system.

2.2 Sequent Calculus

Gentzen's famous sequent calculi **LK** (logischer klassischer Kalkül) and **LJ** (logischer intuitionistischer Kalkül) for classical first-order and intuitionistic logic, respectively, are based on so called sequents. Gentzen's motivation for his sequent calculi was the fact that they allowed him to investigate properties of the calculi of natural deduction in an easier and more elegant way. An important result of his work was the Hauptsatz (or cut-elimination theorem) [9].

A sequent calculus consists of sets of axioms and inference rules that are applied to sequents. Sequents are structures with sequences of formulas on the left and on the right hand side of a symbol (the sequent sign $\vdash$), which does not belong to the syntax of formulas. The definition of sequents is based on [6] Definition 3.1.7.

Definition 2.2.1. Sequent. Let Γ and Δ be finite (possibly empty) multisets of formulas. Then the expression $S : \Gamma \vdash \Delta$ is called a sequent. Γ is called the antecedent of S and Δ the consequent of S. $\vdash$ is also a sequent and is called the empty sequent.

Two sequents $\Gamma_1 \vdash \Delta_1$ and $\Gamma_2 \vdash \Delta_2$ are equal if $\Gamma_1 = \Gamma_2$ and $\Delta_1 = \Delta_2$ [5]. Multiset union within sequents is denoted by comma. If $S : \Gamma \vdash \Delta$ and Γ is the multiset union of Γ_1 and Γ_2 and Δ is the multiset union of Δ_1 and Δ_2, then we write $S : \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2$.

The next definition is based on [6] Definition 3.1.8.

Definition 2.2.2. Semantics of sequents. Consider a sequent

$$S : A_1, ..., A_n \vdash B_1, ..., B_m$$

Then the semantics of S can be expressed by the following PL-formula

$$F(S) : \bigwedge_{i=1}^{n} A_i \rightarrow \bigvee_{j=1}^{m} B_j.$$

$\mathscr{M}$ is an interpretation of S if $\mathscr{M}$ is an interpretation of $F(S)$. If there are no formulas in the antecedent of S (i.e. $n = 0$) we assign $\top$ to $\bigwedge_{i=1}^{n} A_i$. If $m = 0$ we assign $\bot$ to $\bigvee_{j=1}^{m} B_j$. The empty sequent is represented by $\top \rightarrow \bot$ which is equivalent to $\bot$, hence it represents falsum. S is true in $\mathscr{M}$ if $F(S)$ is true in $\mathscr{M}$ and S is called valid if $F(S)$ is valid.

Example 2.2.1. [6] Example 3.1.5. Let

$$S : P(a), (\forall x)(P(x) \rightarrow P(f(x))) \vdash P(f(a))$$

be a sequent. The corresponding formula

$$F(S) : (P(a) \wedge (\forall x)(P(x) \rightarrow P(f(x)))) \rightarrow P(f(a))$$

is valid, so S is a valid sequent.

The following definitions are taken from [6].

Definition 2.2.3. Atomic sequent. A sequent $A_1, ..., A_n \vdash B_1, ..., B_m$ is called atomic if the A_i, B_j for $1 \leq i \leq n$ and $1 \leq j \leq m$ are atomic formulas.

Definition 2.2.4. Composition of sequents. If $S = \Gamma \vdash \Delta$ and $S' = \Pi \vdash \Lambda$ we define the composition of S and S' by $S \circ S'$, where $S \circ S' = \Gamma, \Pi \vdash \Delta, \Lambda$.

The following definition is based on [5] Definition 2.5.

Definition 2.2.5. Subsequent. Let S, S' be sequents. We define $S' \sqsubseteq S$ if there exists a sequent S'' such that $S' \circ S'' = S$ and call S' a subsequent of S.

By definition of the semantics of sequents, every sequent is implied by all of its subsequents. The empty sequent (which stands for falsum) implies every sequent [6].

Definition 2.2.6. [6] Definition 3.1.14. Substitutions can be extended to sequents in an obvious way. If $S = A_1, ..., A_n \vdash B_1, ..., B_m$ and σ is a substitution then

$$S\sigma = A_1\sigma, ..., A_n\sigma \vdash B_1\sigma, ..., B_m\sigma$$

The calculus LK

The next definition is taken from [6] Definition 3.2.1.

Definition 2.2.7. Axiom set. Let $\mathscr{A}$ be a (possibly infinite) set of sequents. $\mathscr{A}$ is called an axiom set if it is closed under substitution, i.e., for all $S \in \mathscr{A}$ and for all substitutions σ it holds that $S\sigma \in \mathscr{A}$. If $\mathscr{A}$ consists only of atomic sequents we have an atomic axiom set.

The closure under substitution is required for proof transformations, in particular for cut-elimination [6].

The following definition is taken from [6] Definition 3.2.2.

Definition 2.2.8. Standard axiom set. Let $\mathscr{A}_T$ be the smallest axiom set containing all sequents of the form $A \vdash A$ for arbitrary atomic formulas A. $\mathscr{A}_T$ is called the standard axiom set.

Definition 2.2.9. LK. Basically we use Gentzen's version of **LK** [9]. Since we consider multisets of formulas, we do not explicitly include exchange or permutation rules. There are two groups of rules, the logical and the structural ones. All rules except the cut have left and right versions, denoted by l and r, respectively. Every logical rule introduces a logical operator on the left or on the right side of a sequent. Structural rules are used to make logical inferences possible or to put proofs together. In the following, A and B denote formulas whereas $\Gamma, \Delta, \Pi, \Lambda$ denote sequences of formulas. Every rule, except weakening and the cut rule, has auxiliary formulas (the formulas in the premises used for the inference) and principal formula (the inferred formula in the conclusion). Weakening has no auxiliary formula and the cut rule has no principal formula.
The logical rules:

- $\land$-introduction

$$\frac{A, \Gamma \vdash \Delta}{A \land B, \Gamma \vdash \Delta}\land_{l_1} \qquad \frac{B, \Gamma \vdash \Delta}{A \land B, \Gamma \vdash \Delta}\land_{l_2} \qquad \frac{\Gamma \vdash \Delta, A \qquad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \land B}\land_r$$

- $\lor$-introduction

$$\frac{A, \Gamma \vdash \Delta \qquad B, \Gamma \vdash \Delta}{A \lor B, \Gamma \vdash \Delta}\lor_l \qquad \frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, A \lor B}\lor_{r_1} \qquad \frac{\Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \lor B}\lor_{r_2}$$

- $\to$-introduction

$$\frac{\Gamma \vdash \Delta, A \qquad B, \Pi \vdash \Lambda}{A \to B, \Gamma, \Pi \vdash \Delta, \Lambda}\to_l \qquad \frac{A, \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \to B}\to_r$$

- $\neg$-introduction

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

- $\forall$-introduction

$$\frac{A\{x \leftarrow t\}, \Gamma \vdash \Delta}{(\forall x)A(x), \Gamma \vdash \Delta}\forall_l \qquad \frac{\Gamma \vdash \Delta, A\{x \leftarrow \alpha\}}{\Gamma \vdash \Delta, (\forall x)A}\forall_r$$

where t is an arbitrary term that does not contain any variables which are bound in A and α is a free variable which may not occur in Γ, Δ, A. α is called an eigenvariable.

- $\exists$-introduction

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

where the variable conditions for $\exists_l$ are the same as those for $\forall_r$ and similarly for $\exists_r$ and $\forall_l$.

The structural rules:

- weakening

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

- contraction

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

- cut
Assume that A occurs in Δ and in Π. Then we define

$$\frac{\Gamma \vdash \Delta \qquad \Pi \vdash \Lambda}{\Gamma, \Pi^* \vdash \Delta^*, \Lambda}\ cut(A)$$

where Π^* is Π after deletion of at least one occurrence of A and Δ^* is Δ after deletion of at least one occurrence of A. A is the auxiliary formula of $cut(A)$ and there is no principal one. A is also called the cut-formula. If the formula A does not occur in Π^* and Δ^* we speak about a *mix* instead of a *cut*. If A is not an atomic formula the cut is called essential, and inessential if A is an atom.

Definition 2.2.10. [6] Definition 3.2.4. Let

$$\frac{S_1 \qquad S_2}{S}\ \xi$$

be a binary rule of **LK** and let S', S'_1, S'_2 be instantiations of the schema variables in S, S_1, S_2. Then (S'_1, S'_2, S') is called an instance of ξ. The instance of a unary rule is defined analogously.

The following definition is based on [6] Definition 3.2.5.

Definition 2.2.11. LK-derivation. An **LK**-derivation is defined by a finite labelled tree with nodes labelled by sequents (via *Seq* function) and edges labelled by the corresponding rule applications. By end-sequent we mean the label of the root and by initial sequents or axioms we mean sequents occurring at the leaves. A formal definition is the following:

- Consider a node ν and an arbitrary sequent S and let $Seq(\nu) = S$. Then ν is an **LK**-derivation and ν is the root node (and also a leaf).
- Let φ be a derivation tree and ν be a leaf in φ and $Seq(\nu) = S$. Let (S_1, S_2, S) be an instance of the binary **LK**-rule ξ. By appending the edges $e_1 : (\nu, \mu_1)$, $e_2 : (\nu, \mu_2)$ to ν such that $Seq(\mu_1) = S_1$ and $Seq(\mu_2) = S_2$ and the label of e_1, e_2 is ξ we extend φ to φ'. φ' is an **LK**-derivation with the same root as φ. μ_1 and μ_2 are leaves in φ' and ν is not, it is called a ξ-node in φ'.
- Let φ be a derivation tree and ν be a leaf in φ s.t. $Seq(\nu) = S$. Let (S', S) be an instance of a unary **LK**-rule ξ. By appending the edge $e : (\nu, \mu)$ to ν such that $Seq(\mu) = S'$ and the label of e is ξ we extend φ to φ'. φ' is an **LK**-derivation with the same root as $\varphi.\mu$ is a leaf in φ' and ν is not, it is called a ξ-node in φ'.

We write

$$\begin{array}{c}(\phi)\\ S\end{array}$$

To express that ϕ is an **LK**-derivation with end sequent S.

The following definition is based on [6] Definition 3.2.9.

Definition 2.2.12. Path. Let φ be an **LK**-derivation and $\pi : \mu_1, ..., \mu_n$ be a sequence of nodes in φ such that for all $i \in \{1, ..., n-1\}$ (μ_i, μ_{i+1}) is an edge in φ. Then π is called a path in φ from μ_1 to μ_n of length $n-1$. We denote the length of a path π by $lp(\pi)$. π is called a trivial path if $n = 1$ and $\pi = \mu_1$. π is called a branch if μ_1 is the root of φ and μ_n is a leaf in φ. The terms predecessor and successor are used contrary to the direction of edges in the tree, if there exists a path from μ_1 to μ_2 then μ_2 is the predecessor of μ_1. The successor relation is defined analogously and every initial sequent is a predecessor of the end sequent.

We follow [6] Definition 3.2.10 by defining a subderivation.

Definition 2.2.13. Subderivation. Let φ' be the subtree of an **LK**-derivation φ with root node ν, where ν is a node in φ. Then φ' is called a subderivation of φ and we write $\varphi' = \varphi.\nu$. Let ρ be an arbitrary **LK**-derivation of $Seq(\nu)$. Then we write $\varphi[\rho]_\nu$ for the deduction φ after the replacement of the subderivation $\varphi.\nu$ by ρ on the node ν in φ, under the restriction that $\varphi.\nu$ and ρ have the same end-sequent.

The following two definitions are taken from [6] Definition 3.2.11 and Definition 3.2.12.

Definition 2.2.14. Depth. Let φ be an **LK**-derivation and ν be a node in φ. Then the depth of ν (denoted by $depth(\nu)$) is defined by the maximal length of a path from ν to a leaf of $\varphi.\nu$. The depth of any leaf in φ is zero.

Definition 2.2.15. Regularity. An **LK**-derivation φ is called regular if

- all eigenvariables of quantifier introductions $\forall_r$ and $\exists_l$ in φ are mutually different.
- If an eigenvariable α occurs as an eigenvariable in a proof node ν then α occurs only above ν in the proof tree.

There exists a straightforward transformation from **LK**-derivations into regular ones, by just renaming the eigenvariables in different subderivations [6]. From now on we assume, without mentioning the fact explicitly, that all **LK**-derivations are regular.

The formulas in sequents on the branch of a deduction tree are connected by a so-called ancestor relation [6]. If A occurs in a sequent S and A is the principal formula of a binary inference on the sequents S_1, S_2, then the auxiliary formulas in S_1, S_2 are immediate ancestors of A. If A occurs in S_1 and is not an auxiliary formula of an inference, then A occurs also in S. In this case S in S_1 is an immediate ancestor of A in S, too. The case of unary rules is analogous. General ancestors are defined via reflexive and transitive closure of the relation.

Definition 2.2.16. [6] Definition 3.2.14. Let Ω be a set of formula occurrences in an **LK**-derivation φ and ν be a node in φ. Then $S(\nu, \Omega)$ is the subsequent of $Seq(\nu)$ obtained by deleting all formula occurrences which are not ancestors of occurrences on Ω.

If Ω consists just of the occurrences of all cut formulas which occur below ν then $S(\nu, \Omega)$ is the subsequent of $Seq(\nu)$ consisting of all formulas which are ancestors of a cut. These subsequents are crucial for the definition of the characteristic set of clauses and of the method CERES [6].

Definition 2.2.17. Proof length. The length of a proof φ is defined by the number of nodes in φ. We denote the proof length of a proof φ by $l(\varphi)$.

The following two definitions are based on [6] Definition 3.2.16 and Definition 3.2.17.

Definition 2.2.18. Cut-derivation. Let φ be an **LK**-derivation of the following form

$$\frac{\begin{array}{cc}(\varphi_1) & (\varphi_2) \\ \Gamma_1 \vdash \Delta_1, A & A, \Gamma_2 \vdash \Delta_2\end{array}}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2}\; cut(A)$$

Then φ is called a cut-derivation. φ_1 and φ_2 may contain cuts, too. If the cut is a mix we speak about a mix-derivation. φ is called essential if the cut is essential.

Definition 2.2.19. Rank, grade. Let φ be an **LK**-derivation of the following form

$$\frac{\begin{array}{cc}(\varphi_1) & (\varphi_2) \\ \Gamma_1 \vdash \Delta_1, A & A, \Gamma_2 \vdash \Delta_2\end{array}}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2}\; cut(A)$$

Then the grade of φ is $comp(A)$.
Let μ and ν be the root nodes of φ_1 and φ_2, respectively. An A-right path in φ_1 is a path in φ_1 of the form $\mu, \mu_1, ..., \mu_n$ such that A occurs in the consequents of all $Seq(\mu_i)$. Similarly an A-left path in φ_2 is a path in φ_2 of the form $\nu, \nu_1, ..., \nu_m$ such that A occurs in the antecedents of all $Seq(\nu_j)$. Let P_1 be the set of all A-right paths in φ_1 and P_2 the set of all A-left paths in φ_2. Then we define the left-rank of φ ($rank_l(\varphi)$) and the right-rank ($rank_r(\varphi)$) as

$$rank_l(\varphi) = max\{lp(\pi) \| \pi \in P_1\} + 1,$$

$$rank_r(\varphi) = max\{lp(\pi) \| \pi \in P_2\} + 1.$$

The rank if φ is defined as $rank(\varphi) = rank_l(\varphi) + rank_r(\varphi)$.

2.3 Resolution Calculus

The resolution calculus was introduced by Robinson in 1965 [17]. It was specifically designed as a theoretical basis to be used in automated theorem proving. Robinson's principle lead to enormous improvements in performance compared to prior methods [14]. The resolution calculus is a so-called refutation calculus, the goal is to refute a statement instead of proving that it is a theorem.

Our formulation of the resolution calculus is based on sets of specific sequents and uses most general unification as well as the rules of resolution.

The following definition is taken from [5] Definition 2.12.

Definition 2.3.1. Clause. A clause is an atomic sequent, i.e. a sequent of the form $\Gamma \vdash \Delta$ where Γ and Δ are mutlisets of atomic formulas.

Clauses are usually defined as disjunctions of literals. A literal is either an atom or a negated atom. We follow [6] Definition 3.3.1 by defining a unifier and a most general unifier. Most general unification was the key novel feature of the resolution principle by Robinson [17] where he proved that for all unifiable sets there exists also a most general unifier, which makes the computation of other unifiers superfluous.

Definition 2.3.2. Unifier. Let $\mathscr{A}$ be a nonempty set of atoms and σ be a substitution. σ is called a unifier of $\mathscr{A}$ if the set $\mathscr{A}\sigma$ contains only one element. σ is called a most general unifier (or m.g.u.) of $\mathscr{A}$ if σ is a unifier of $\mathscr{A}$ and for all unifiers λ of $\mathscr{A}$ it holds that $\sigma \leq_s \lambda$.

The following theorem correspond to [14] Theorem 2.6.1.

Theorem 2.3.1. *Unification Theorem. There exists a decision procedure UAL for the unifiability of two terms. In particular, the following two properties hold:*

- *If $\{t_1, t_2\}$ is not unifiable, then UAL stops with failure.*

- *If $\{t_1, t_2\}$ is unifiable, then UAL stops and σ (the final substitution constructed by UAL) is a most general unifier of $\{t_1, t_2\}$.*

Proof. See proof of Theorem 2.6.1 in [14]. □

The following definitions are based on [6] Definition 3.3.10, Definition 3.3.11 and Definition 3.3.12.

Definition 2.3.3. Resolvent. Let C and D be clauses of the form

$$C = \Gamma \vdash \Delta_1, A_1, ..., \Delta_n, A_n, \Delta_{n+1},$$

$$D = \Pi_1, B_1, ..., \Pi_m, B_m, \Pi_{m+1} \vdash \Lambda$$

such that C and D do not share variables and the set $\{A_1, ..., A_n, B_1, ..., B_m\}$ is unifiable by a most general unifier σ. Then the clause

$$R : \Gamma\sigma, \Pi_1\sigma, ..., \Pi_{m+1}\sigma \vdash \Delta_1\sigma, ..., \Delta_{n+1}\sigma, \Lambda\sigma$$

is called a resolvent of C and D.

Definition 2.3.4. P-resolvent. Let $C = \Sigma \vdash \Delta, A^m$ and $D = A^n, \Pi \vdash \Lambda$ with $n, m \geq 1$. Then the clause $\Gamma, \Pi \vdash \Delta, \Lambda$ is called a p-resolvent of C and D.

The p-resolvents of C and D are the sequents obtained by applying the cut rule to C and D. Therefore resolution of clauses is a cut combined with most general unification.

When we want to resolve two clauses C_1, C_2 we have to ensure that C_1 and C_2 are variable disjoint. This can be achieved by renaming variables by permutation of variables.

Definition 2.3.5. Variant. Let C be a clause and σ a permutation substitution (i.e. π is a binary function $V \to V$). Then $C\sigma$ is called a variant of C.

The following definition is based on [6] Definition 3.3.13.

Definition 2.3.6. Resolution deduction. Consider a labelled tree γ like an **LK**-derivation with the exception that it is binary and all edges are labelled by the resolution rule. Then γ is a resolution deduction. If we replace the resolutions by p-resolutions we speak about a p-resolution deduction. A ground resolution deduction is a p-resolution deduction, where all clauses are variable-free. Let $\mathscr{C}$ be a set of clauses. If all initial sequents in γ are variants of clauses in $\mathscr{C}$ and D is the clause labelling the root, then γ is called a resolution derivation of D from $\mathscr{C}$. If $D = \vdash$ then γ is called a resolution refutation of $\mathscr{C}$.

The next definition is taken from [6] Definition 3.3.14.

Definition 2.3.7. Ground projection. Let γ' be a ground resolution deduction which is an instance of a resolution deduction γ. Then γ' is called a ground projection of γ.

Theorem 2.3.2. *Completeness of resolution deduction. If* $\mathscr{C}$ *is an unsatisfiable set of clauses, then there exists a resolution refutation of* $\mathscr{C}$.

Proof. By Theorem 2.7.2 in [14]. □

CHAPTER 3

The Problem of Cut-Elimination

This chapter is intended to give an overview on cut-elimination as introduced by Gentzen in his seminal papers [9]. In Section 3.1 we will give a motivation on cut-elimination, in Section 3.2 we will deal with cut-elimination and its consequences in a more formal way and in Section 3.3 we will conclude this chapter by defining a rewriting system for cut-elimination based on the rules obtained from Gentzen's original proof of the cut-elimination theorem.

3.1 Motivation

Cut-Elimination is a constructive method for proving the so-called Hauptsatz (or cut-elimination theorem) for **LK** and **LJ** [6], [9] and was introduced by Gentzen. It states that any theorem of first-order logic can be proved without detours, i.e. without the use of cuts.

Basically, cut-elimination is concerned with the elimination of all cuts from a proof. Since cuts correspond to the use of lemmas (i.e. intermediary statements) in mathematical proofs, the elimination of cuts corresponds to the elimination of lemmas. Therefore the cut-elimination theorem implies that any statement can be proved without the use of lemmas [5].

Cut-free proofs have the property that all formulas used in the proof are (instances of) subformulas of the end-sequent, i.e. they have the subformula property [5]. This leads to one of the most important consequences of the cut-elimination theorem, the consistency of both **LK** and **LJ**. Indeed, if there was a proof of the empty sequent, then it would be provable without cuts, which is impossible due to the subformula property of cut-free proofs.

It was shown in Gentzen's sharpened Hauptsatz (or midsequent theorem) that in a cut-free proof of a sequent, which contains only formulas in prenex form, there exists a so-called midsequent. The midsequent splits the proof into an upper part, containing the propositional inferences and into a lower part, containing the quantifier inferences [9]. This allows the extraction of Herbrand sequents.

The cut-elimination theorem can also be used as a method of proof mining in the sense that hidden mathematical information can be extracted by eliminating lemmas from proofs [15]. The extraction of functionals from proofs, which is another approach for proof mining, is based on Gödel's dialectica interpretation [12]. It allows the construction of programs from proofs [15]. Functional interpretation can also be used for the extraction of Herbrand disjunctions from proofs [10].

Cut-elimination can be applied to proofs in real mathematics as well. By applying cut-elimination to Fürstenberg and Weiss' topological proof [8] of van der Waerdens' theorem [21] and eliminating all lemmas contained in the proof, Girard was able to obtain van der Waerden's original proof as a result [11].

3.2 Cut-Elimination Theorem

Theorem 3.2.1. *[9], Gentzen 1934. If a sequent is **LK**-provable, then it is **LK**-provable without a cut.*

Proof. We will only give an outline of the proof, for the full proof we refer to [9]. Consider an **LK**-proof φ of some arbitrary end-sequent S. The proof is by double induction on $grade(\varphi)$ and $rank(\varphi)$. The uppermost cut (in fact the uppermost mix) is eliminated by permuting the cut upwards and thus reducing the rank until no longer possible. This is the case when the cut occurs immediately below the inferences that introduced its cut-formula in both premises. Then the grade of the cut-formula A is reduced by replacing this cut by cuts, where the cut-formulas are subformulas of A. Cuts that have axioms as premises can be eliminated completely. Iterating this procedure eventually yields a cut-free proof of he same end-sequent S. □

Corollary. *[20], Theorem 6.3 In a cut-free proof in **LK** (or **LJ**) all the formulas which occur in it are subformulas of the formulas in the end-sequent.*

Proof. By mathematical induction on the number of inferences in the cut-free proof. □

The following corollary corresponds to [20], Theorem 6.2.

Corollary. *Consistency. **LK** and **LJ** are consistent.*

Proof. Assume $\vdash$ was provable in **LK** (or **LJ**). Then by Theorem 3.1 it is provable in **LK** (or **LJ**) without a cut. But this is impossible by the subformula property of cut-free proofs. □

The following theorem is a formulation of Gentzen's midsequent theorem.

Theorem 3.2.2. *[20], Theorem 6.4. Let S be a sequent which consists of prenex formulas only and is provable in **LK**. Then there is a cut-free proof of S which contains a sequent called midsequent S', which satisfies the following properties:*

- *S' is quantifier-free,*
- *every inference above S' is either structural or propositional and*
- *every inference below S' is either structural or a quantifier inference.*

Proof. See [20]. □

3.3 Reductive Cut-Elimination

Gentzen's proof of the Hauptsatz [9] already contains an algorithm for removing all cuts from an **LK**-proof. This method is often referred to as reductive cut-elimination, because it is based on proof-rewriting rules that reduce the complexity of the cut-formula one by one until atomic cuts are reached which can be removed completely. The transformation steps from an **LK**-proof of some sequent S with cuts into an **LK**-proof without cuts of the same sequent can be used to define a proof rewriting system whose normal forms are cut-free proofs [5]. Such a rewrite system (or also called a reduction system) will be described in more detail in the following section.

Another method for cut-elimination which is closely related to Gentzen's procedure is the method of Schütte [18] and Tait [19]. This method eliminates an uppermost cut in a proof whose cut-formula has maximal complexity, i.e. if the cut-formula of the uppermost cut is A then $comp(B) \leq comp(A)$ for all other cut-formulas B in the proof [5].

The following definition is based on [5].

Definition 3.3.1. Cut-reduction system. Let Φ be the set of all **LK**-derivations. The pair $\mathscr{R} = \langle \Phi, >_{\mathscr{R}} \rangle$ is called a cut-reduction system where $>_{\mathscr{R}} \subseteq \Phi \times \Phi$ is a binary relation over **LK**-derivations. Assume $\varphi, \phi \in \Phi$, then $\varphi >_{\mathscr{R}} \phi$ if and only if φ reduces to ϕ according to the cut-reduction rules specified in Definition 3.3.7.

Definition 3.3.2. [5] Definition 3.1. Let $> \subseteq \Phi \times \Phi$. We say that $>$ is based on $\mathscr{R} = \langle \Phi, >_{\mathscr{R}} \rangle$ if $> \subseteq >_{\mathscr{R}}$ and write $\phi > \varphi$ for $(\phi, \varphi) \in >$. Analogous for $\mathscr{R}_{ax}$.

Definition 3.3.3. [5] Definition 3.2. Let $\phi, \varphi \in \Phi$ with $\phi >_{\mathscr{R}} \varphi$ and let $\pi \in \Phi$ such that $\pi.\nu = \phi$ for a node ν in π. Then we define $\pi >_{\mathscr{R}} \pi[\varphi]_\nu$, i.e. $>_{\mathscr{R}}$ is closed under contexts. Analogous for $\mathscr{R}_{ax}$.

The following definitions are taken from [5] Definition 3.3 and Definition 3.4, respectively.

Definition 3.3.4. Gentzen reduction. We define $\phi >_G \pi$ if $\phi >_{\mathscr{R}} \pi$ and ϕ is a cut-derivation with a single non-atomic cut, which is the last inference. $>_G$ is extended like $>_{\mathscr{R}}$: $\varphi >_G \varphi'$ if $\varphi' = \varphi[\pi]_\nu$ and $\varphi.\nu >_G \pi$.

Definition 3.3.5. Tait reduction. We define $\varphi >_T \varphi'$ if the following conditions are fulfilled:

- There exists a node ν in φ such that $\varphi.\nu$ is a cut-derivation with a maximal cut-formula, i.e. if the cut-formula of the last cut in $\varphi.\nu$ is A then $comp(B) \leq comp(A)$ for all other cut-formulas B in φ.
- $\varphi.\nu$ is strict, i.e. for all other cut-formulas B in $\varphi.\nu$ we have $comp(B) < comp(A)$.
- $\varphi' = \varphi[\pi]_\nu$ for an **LK**-derivation π with $\varphi.\nu >_{\mathscr{R}} \pi$.

The definition of an atomic cut normal form is based on [5] Definition 3.5.

Definition 3.3.6. ACNF. Let $>$ be a cut-reduction relation based on $\mathscr{R}$. Then an **LK**-derivation ϕ is in atomic cut normal form (ACNF) w.r.t. $>$ if there exists no π such that $\phi > \pi$.

Consider $>^*$, the reflexive and transitive closure of $>$. We say that ϕ is an ACNF of φ if ϕ is in ACNF and $\varphi >^* \phi$.

Cut-reduction rules Now we will define the cut-reduction rules. They can be divided into cut-elimination, grade reduction and rank reduction rules. With grade reductions a cut with non-atomic cut-formulas is replaced by a new cut whose cut-formulas are subformulas of the non-atomic cut-formulas. Rank reductions are used to permute a cut over unary or binary rules upwards in the proof. Cut-eliminaiton rules are used to transform a proof ϕ into a proof ϕ' s.t. ϕ' is the result of eliminating a cut in ϕ, i.e. for Gentzen's method this means that the uppermost cut is eliminated.

The following definition is based on [5], [23] and [16].

Definition 3.3.7. Cut-reduction Rules
Cut-elimination rules:

- over axioms

$$\dfrac{A \vdash A \qquad \overset{(\sigma)}{A, \Gamma \vdash \Delta}}{A, \Gamma \vdash \Delta}\, cut(A) \qquad\qquad \dfrac{\overset{(\rho)}{\Gamma \vdash \Delta, A} \qquad A \vdash A}{\Gamma \vdash \Delta, A}\, cut(A)$$

$$\Downarrow \qquad\qquad \Downarrow$$

$$\overset{(\sigma)}{A, \Gamma \vdash \Delta} \qquad\qquad \overset{(\rho)}{\Gamma \vdash \Delta, A}$$

- over weakening

$$\dfrac{\dfrac{\overset{(\rho')}{\Gamma \vdash \Delta}}{\Gamma \vdash \Delta, A}\, w_r \qquad \overset{(\sigma)}{A, \Pi \vdash \Lambda}}{\Gamma, \Pi \vdash \Delta, \Lambda}\, cut(A) \qquad\qquad \dfrac{\overset{(\rho)}{\Gamma \vdash \Delta, A} \qquad \dfrac{\overset{(\sigma')}{\Pi \vdash \Lambda}}{A, \Pi \vdash \Lambda}\, w_l}{\Gamma, \Pi \vdash \Delta, \Lambda}\, cut(A)$$

$$\Downarrow \qquad\qquad \Downarrow$$

$$\dfrac{\begin{matrix}(\rho')\\ \Gamma \vdash \Delta\end{matrix}}{\Gamma, \Pi \vdash \Delta, \Lambda}\, w_r^*, w_l^* \qquad\qquad \dfrac{\begin{matrix}(\sigma')\\ \Pi \vdash \Lambda\end{matrix}}{\Gamma, \Pi \vdash \Delta, \Lambda}\, w_r^*, w_l^*$$

Grade reduction rules:

- cut-formula has a $\neg$ as top-level connective

$$\dfrac{\dfrac{\begin{matrix}(\rho')\\ A, \Gamma \vdash \Delta\end{matrix}}{\Gamma \vdash \Delta, \neg A}\,\neg_r \qquad \dfrac{\begin{matrix}(\sigma')\\ \Pi \vdash \Lambda, A\end{matrix}}{\neg A, \Pi \vdash \Lambda}\,\neg_l}{\Gamma, \Pi \vdash \Delta, \Lambda}\, cut(\neg A)$$

$$\Downarrow$$

$$\dfrac{\begin{matrix}(\sigma')\\ \Pi \vdash \Lambda, A\end{matrix} \qquad \begin{matrix}(\rho')\\ A, \Gamma \vdash \Delta\end{matrix}}{\Gamma, \Pi \vdash \Delta, \Lambda}\, cut(A)$$

- cut-formula has a $\wedge$ as top-level connective

$$\dfrac{\dfrac{\begin{matrix}(\rho_1)\\ \Gamma \vdash \Delta, A_1\end{matrix} \qquad \begin{matrix}(\rho_2)\\ \Gamma \vdash \Delta, A_2\end{matrix}}{\Gamma \vdash \Delta, A_1 \wedge A_2}\,\wedge_r \qquad \dfrac{\begin{matrix}(\sigma')\\ A_i, \Pi \vdash \Lambda\end{matrix}}{A_1 \wedge A_2, \Pi \vdash \Lambda}\,\wedge_{l_i}}{\Gamma, \Pi \vdash \Delta, \Lambda}\, cut(A_1 \wedge A_2)$$

$$\Downarrow$$

$$\dfrac{\dfrac{\begin{matrix}(\rho_1)\\ \Gamma \vdash \Delta, A_1\end{matrix} \qquad \dfrac{\begin{matrix}(\rho_2)\\ \Gamma \vdash \Delta, A_2\end{matrix} \qquad \dfrac{\begin{matrix}(\sigma')\\ A_i, \Pi \vdash \Lambda\end{matrix}}{A_1, A_2 \Pi \vdash \Lambda}\, w_l}{A_1, \Gamma, \Pi \vdash \Delta, \Lambda}\, cut(A_2)}{\Gamma, \Gamma, \Pi \vdash \Delta, \Delta, \Lambda}\, cut(A_1)}{\Gamma, \Pi \vdash \Delta, \Lambda}\, c_l^*, c_r^*$$

- cut-formula has a $\vee$ as top-level connective

$$\dfrac{\dfrac{\begin{matrix}(\rho)\\ \Gamma \vdash \Delta, A_i\end{matrix}}{\Gamma \vdash \Delta, A_1 \vee A_2}\,\vee_{r_i} \qquad \dfrac{\begin{matrix}(\sigma_1)\\ A_1, \Pi \vdash \Lambda\end{matrix} \qquad \begin{matrix}(\sigma_2)\\ A_2, \Pi \vdash \Lambda\end{matrix}}{A_1 \vee A_2, \Pi \vdash \Lambda}\,\vee_l}{\Gamma, \Pi \vdash \Delta, \Lambda}\, cut(A_1 \vee A_2)$$

$$\Downarrow$$

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}(\rho)\\ \Gamma\vdash\Delta,A_i\end{array}}{\Gamma\vdash\Delta,A_1,A_2}\,w_r \qquad \begin{array}{c}(\sigma_2)\\ A_2,\Pi\vdash\Lambda\end{array}}{\Gamma,\Pi\vdash\Delta,\Lambda,A_1}\,cut(A_2) \qquad \begin{array}{c}(\sigma_1)\\ A_1,\Pi\vdash\Lambda\end{array}}{\dfrac{\Gamma,\Pi,\Pi\vdash\Delta,\Lambda,\Lambda}{\Gamma,\Pi\vdash\Delta,\Lambda}\,c_l^*,c_r^*}\,cut(A_1)$$

- cut-formula has a $\exists$ as top-level connective

$$\dfrac{\dfrac{\begin{array}{c}(\rho')\\ \Gamma\vdash\Delta,A(x/t)\end{array}}{\Gamma\vdash\Delta,(\exists x)A(x)}\,\exists_r \qquad \dfrac{\begin{array}{c}(\sigma'(x/y))\\ A(x/y),\Pi\vdash\Lambda\end{array}}{(\exists x)A(x),\Pi\vdash\Lambda}\,\exists_l}{\Gamma,\Pi\vdash\Delta,\Lambda}\,cut((\exists x)A)$$

$$\Downarrow$$

$$\dfrac{\begin{array}{c}(\rho')\\ \Gamma\vdash\Delta,A(x/t)\end{array} \qquad \begin{array}{c}(\sigma'(x/t))\\ A(x/t),\Pi\vdash\Lambda\end{array}}{\Gamma,\Pi\vdash\Delta,\Lambda}\,cut(a(x/t))$$

- cut-formula has a $\forall$ as top-level connective

$$\dfrac{\dfrac{\begin{array}{c}(\rho'(x/y))\\ \Gamma\vdash\Delta,A(x/y)\end{array}}{\Gamma\vdash\Delta,(\forall x)A(x)}\,\forall_r \qquad \dfrac{\begin{array}{c}(\sigma')\\ A(x/t),\Pi\vdash\Lambda\end{array}}{(\forall x)A(x),\Pi\vdash\Lambda}\,\forall_l}{\Gamma,\Pi\vdash\Delta,\Lambda}\,cut((\forall x)A)$$

$$\Downarrow$$

$$\dfrac{\begin{array}{c}(\rho'(x/t))\\ \Gamma\vdash\Delta,A(x/t)\end{array} \qquad \begin{array}{c}(\sigma')\\ A(x/t),\Pi\vdash\Lambda\end{array}}{\Gamma,\Pi\vdash\Delta,\Lambda}\,cut(a(x/t))$$

Rank reduction rules:

- over a unary rule Ξ

$$\dfrac{\dfrac{\overset{(\rho')}{\Gamma' \vdash \Delta', A}}{\Gamma \vdash \Delta, A}\,\Xi \qquad \overset{(\sigma)}{A, \Pi \vdash \Lambda}}{\Gamma, \Pi \vdash \Delta, \Lambda}\,cut(A) \qquad\qquad \dfrac{\overset{(\rho)}{\Gamma \vdash \Delta, A} \qquad \dfrac{\overset{(\sigma')}{A, \Pi' \vdash \Lambda'}}{A, \Pi \vdash \Lambda}\,\Xi}{\Gamma, \Pi \vdash \Delta, \Lambda}\,cut(A)$$

$$\Downarrow \qquad\qquad\qquad\qquad \Downarrow$$

$$\dfrac{\dfrac{\overset{(\rho')}{\Gamma' \vdash \Delta', A} \qquad \overset{(\sigma)}{A, \Pi \vdash \Lambda}}{\Gamma', \Pi \vdash \Delta', \Lambda}\,cut(A)}{\Gamma, \Pi \vdash \Delta, \Lambda}\,\Xi \qquad\qquad \dfrac{\dfrac{\overset{(\rho)}{\Gamma \vdash \Delta, A} \qquad \overset{(\sigma')}{A, \Pi' \vdash \Lambda'}}{\Gamma, \Pi' \vdash \Delta, \Lambda'}\,cut(A)}{\Gamma, \Pi \vdash \Delta, \Lambda}\,\Xi$$

- over a binary rule Ξ

$$\dfrac{\dfrac{\overset{(\rho_1)}{\Gamma_1 \vdash \Delta_1, A} \qquad \overset{(\rho_2)}{\Gamma_2 \vdash \Delta_2}}{\Gamma \vdash \Delta, A}\,\Xi \qquad \overset{(\sigma)}{A, \Pi \vdash \Lambda}}{\Gamma, \Pi \vdash \Delta, \Lambda}\,cut(A)$$

$$\Downarrow$$

$$\dfrac{\dfrac{\dfrac{\overset{(\rho_1)}{\Gamma_1 \vdash \Delta_1, A} \qquad \overset{(\sigma)}{A, \Pi \vdash \Lambda}}{\Gamma_1, \Pi \vdash \Delta_1, \Lambda}\,cut(A) \qquad \dfrac{\dfrac{\overset{(\rho_2)}{\Gamma_2 \vdash \Delta_2}}{\Gamma_2 \vdash \Delta_2, A}\,w_r \qquad \overset{(\sigma')}{A, \Pi \vdash \Lambda}}{\Gamma_2, \Pi \vdash \Delta_2, \Lambda}\,cut(A)}{\Gamma, \Pi, \Pi \vdash \Delta, \Lambda, \Lambda}\,\Xi}{\Gamma, \Pi \vdash \Delta, \Lambda}\,c_l^*, c_r^*$$

$$\dfrac{\dfrac{\overset{(\rho_1)}{\Gamma_1 \vdash \Delta_1} \qquad \overset{(\rho_2)}{\Gamma_2 \vdash \Delta_2, A}}{\Gamma \vdash \Delta, A}\,\Xi \qquad \overset{(\sigma)}{A, \Pi \vdash \Lambda}}{\Gamma, \Pi \vdash \Delta, \Lambda}\,cut(A)$$

$$\Downarrow$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\overset{(\rho_1)}{\Gamma_1 \vdash \Delta_1}}{\Gamma_1 \vdash \Delta_1, A}\,w_r \qquad \overset{(\sigma)}{A, \Pi \vdash \Lambda}}{\Gamma_1, \Pi \vdash \Delta_1, \Lambda}\,cut(A) \qquad \dfrac{\overset{(\rho_2)}{\Gamma_2 \vdash \Delta_2, A} \qquad \overset{(\sigma')}{A, \Pi \vdash \Lambda}}{\Gamma_2, \Pi \vdash \Delta_2, \Lambda}\,cut(A)}{\Gamma, \Pi, \Pi \vdash \Delta, \Lambda, \Lambda}\,\Xi}{\Gamma, \Pi \vdash \Delta, \Lambda}\,c_l^*, c_r^*$$

where in the above two reductions σ' is obtained from σ by renaming of the eigenvariables such that the regularity of ϕ' is ensured.

$$\dfrac{\genfrac{}{}{0pt}{}{(\rho)}{\Gamma \vdash \Delta, A} \qquad \dfrac{\genfrac{}{}{0pt}{}{(\sigma_1)}{A, \Pi_1 \vdash \Lambda_1} \qquad \genfrac{}{}{0pt}{}{(\sigma_2)}{A, \Pi_2 \vdash \Lambda_2}}{A, \Pi \vdash \Lambda}\,\Xi}{\Gamma, \Pi \vdash \Delta, \Lambda}\,cut(A)$$

$$\Downarrow$$

$$\dfrac{\dfrac{\dfrac{\genfrac{}{}{0pt}{}{(\rho)}{\Gamma \vdash \Delta, A} \qquad \genfrac{}{}{0pt}{}{(\sigma_1)}{A, \Pi_1 \vdash \Lambda_1}}{\Gamma, \Pi_1 \vdash \Delta, \Lambda_1}\,cut(A) \qquad \dfrac{\genfrac{}{}{0pt}{}{(\rho')}{\Gamma \vdash \Delta, A} \qquad \dfrac{\genfrac{}{}{0pt}{}{(\sigma_2)}{\Pi_2 \vdash \Lambda_2}}{A, \Pi_2 \vdash \Lambda_2}\,w_l}{\Gamma, \Pi_2 \vdash \Delta, \Lambda_2}\,cut(A)}{\Gamma, \Gamma, \Pi \vdash \Delta, \Delta, \Lambda}\,\Xi}{\Gamma, \Pi \vdash \Delta, \Lambda}\,c_l^*, c_r^*$$

$$\dfrac{\genfrac{}{}{0pt}{}{(\rho)}{\Gamma \vdash \Delta, A} \qquad \dfrac{\genfrac{}{}{0pt}{}{(\sigma_1)}{\Pi_1 \vdash \Lambda_1} \qquad \genfrac{}{}{0pt}{}{(\sigma_2)}{A, \Pi_2 \vdash \Lambda_2}}{A, \Pi \vdash \Lambda}\,\Xi}{\Gamma, \Pi \vdash \Delta, \Lambda}\,cut(A)$$

$$\Downarrow$$

$$\dfrac{\dfrac{\dfrac{\genfrac{}{}{0pt}{}{(\rho)}{\Gamma \vdash \Delta, A} \qquad \dfrac{\genfrac{}{}{0pt}{}{(\sigma_1)}{\Pi_1 \vdash \Lambda_1}}{A, \Pi_1 \vdash \Lambda_1}\,w_l}{\Gamma, \Pi_1 \vdash \Delta, \Lambda_1}\,cut(A) \qquad \dfrac{\genfrac{}{}{0pt}{}{(\rho')}{\Gamma \vdash \Delta, A} \qquad \genfrac{}{}{0pt}{}{(\sigma_2)}{A, \Pi_2 \vdash \Lambda_2}}{\Gamma, \Pi_2 \vdash \Delta, \Lambda_2}\,cut(A)}{\Gamma, \Gamma, \Pi \vdash \Delta, \Delta, \Lambda}\,\Xi}{\Gamma, \Pi \vdash \Delta, \Lambda}\,c_l^*, c_r^*$$

where in the above two reductions ρ' is obtained from ρ by renaming the eigenvariables such that the regularity of ϕ' is ensured.

- over a contraction rule (contraction right)

$$\dfrac{\dfrac{\genfrac{}{}{0pt}{}{(\rho')}{\Gamma \vdash \Delta, A, A}}{\Gamma \vdash \Delta, A}\,c_r \qquad \genfrac{}{}{0pt}{}{(\sigma)}{A, \Pi \vdash \Lambda}}{\Gamma, \Pi \vdash \Delta, \Lambda}\,cut(A)$$

$$\Downarrow$$

$$\dfrac{\dfrac{\overset{(\rho')}{\Gamma \vdash \Delta, A, A} \quad \overset{(\sigma)}{A, \Pi \vdash \Lambda}}{\Gamma, \Pi \vdash \Delta, \Lambda, A}\, cut(A) \quad \overset{(\sigma')}{A, \Pi \vdash \Lambda}}{\dfrac{\Gamma, \Pi, \Pi \vdash \Delta, \Lambda, \Lambda}{\Gamma, \Pi \vdash \Delta, \Lambda}\, c_r^*, c_l^*}\, cut(A)$$

where σ' is obtained from σ by renaming the eigenvariables such that the regularity of ϕ' is ensured.

- over a contraction rule (contraction left)

$$\dfrac{\overset{(\rho)}{\Gamma \vdash \Delta, A} \quad \dfrac{\overset{(\sigma')}{A, A, \Pi \vdash \Lambda}}{A, \Pi \vdash \Lambda}\, c_l}{\Gamma, \Pi \vdash \Delta, \Lambda}\, cut(A)$$

$$\Downarrow$$

$$\dfrac{\overset{(\rho')}{\Gamma \vdash \Delta, A} \quad \dfrac{\overset{(\rho)}{\Gamma \vdash \Delta, A} \quad \overset{(\sigma')}{A, A, \Pi \vdash \Lambda}}{A, \Gamma, \Pi \vdash \Delta, \Lambda}\, cut(A)}{\dfrac{\Gamma, \Gamma, \Pi \vdash \Delta, \Delta, \Lambda}{\Gamma, \Pi \vdash \Delta, \Lambda}\, c_r^*, c_l^*}\, cut(A)$$

where ρ' is obtained from ρ by renaming the eigenvariables such that the regularity of ϕ' is ensured.

Example 3.3.1. Let φ be the derivation

$$\dfrac{\overset{(\varphi_1)}{P(a) \vee Q(b) \vdash (\exists y)(P(y) \vee Q(y))} \quad \overset{(\varphi_2)}{(\exists y)(P(y) \vee Q(y)), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}}{P(a) \vee Q(b), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\, cut$$

where φ_1 is the **LK**-derivation:

$$\dfrac{\dfrac{\dfrac{P(a) \vdash P(a)}{P(a) \vdash P(a) \vee Q(a)}\, \vee_{r_1}}{P(a) \vdash (\exists y)(P(y) \vee Q(y))}\, \exists_r \quad \dfrac{\dfrac{Q(b) \vdash Q(b)}{Q(b) \vdash P(b) \vee Q(b)}\, \vee_{r_2}}{Q(b) \vdash (\exists y)(P(y) \vee Q(y))}\, \exists_r}{P(a) \vee Q(b) \vdash (\exists y)(P(y) \vee Q(y))}\, \vee_l$$

and φ_2 is the **LK**-derivation:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(u)\vdash P(u)}{P(u),\neg P(u)\vdash}\,\neg_l}{P(u),\neg P(u)\vdash Q(u)}\,w_r \qquad \dfrac{Q(u)\vdash Q(u)}{Q(u),\neg P(u)\vdash Q(u)}\,w_l}{P(u)\lor Q(u),\neg P(u)\vdash Q(u)}\,\lor_l}{P(u)\lor Q(u),\neg P(u)\vdash(\exists z)Q(z)}\,\exists_r}{P(u)\lor Q(u),(\forall x)\neg P(x)\vdash(\exists z)Q(z)}\,\forall_l}{(\exists y)(P(y)\lor Q(y)),(\forall x)\neg P(x)\vdash(\exists z)Q(z)}\,\exists_l$$

For φ we obtain the following cut-reduction sequence:

$$\dfrac{\dfrac{\dfrac{\dfrac{P(a)\vdash P(a)}{P(a)\vdash P(a)\lor Q(a)}\,\lor_{r_1}}{P(a)\vdash(\exists y)(P(y)\lor Q(y))}\,\exists_r \qquad \dfrac{\dfrac{Q(b)\vdash Q(b)}{Q(b)\vdash P(b)\lor Q(b)}\,\lor_{r_2}}{Q(b)\vdash(\exists y)(P(y)\lor Q(y))}\,\exists_r}{P(a)\lor Q(b)\vdash(\exists y)(P(y)\lor Q(y))}\,\lor_l \qquad (\varphi_2)}{P(a)\lor Q(b),(\forall x)\neg P(x)\vdash(\exists z)Q(z)}\,cut$$

via rank-reduction over $\lor_l$ we get

$$\dfrac{\begin{matrix}(\varphi_1')\\ P(a),(\forall x)\neg P(x)\vdash(\exists z)Q(z)\end{matrix} \qquad \begin{matrix}(\varphi_2')\\ Q(b),(\forall x)\neg P(x)\vdash(\exists z)Q(z)\end{matrix}}{P(a)\lor Q(b),(\forall x)\neg P(x)\vdash(\exists z)Q(z)}\,\lor_l$$

where φ_1' is the **LK**-derivation

$$\dfrac{\dfrac{\dfrac{P(a)\vdash P(a)}{P(a)\vdash P(a)\lor Q(a)}\,\lor_{r_1}}{P(a)\vdash(\exists y)(P(y)\lor Q(y))}\,\exists_r \qquad (\varphi_2)}{P(a),(\forall x)\neg P(x)\vdash(\exists z)Q(z)}\,cut$$

and φ_2' is the **LK**-derivation

$$\dfrac{\dfrac{\dfrac{Q(b)\vdash Q(b)}{Q(b)\vdash P(b)\lor Q(b)}\,\lor_{r_2}}{Q(b)\vdash(\exists y)(P(y)\lor Q(y))}\,\exists_r \qquad (\varphi_2'')}{Q(b),(\forall x)\neg P(x)\vdash(\exists z)Q(z)}\,cut$$

where ϕ_2'' is the **LK**-derivation obtained from φ_2 by replacing the eigenvariable u by the eigenvariable v.

First the cut occurring in φ_1' will be eliminated:

$$\dfrac{\dfrac{\dfrac{P(a)\vdash P(a)}{P(a)\vdash P(a)\lor Q(a)}\,\lor_{r_1}}{P(a)\vdash(\exists y)(P(y)\lor Q(y))}\,\exists_r \qquad (\varphi_2)}{P(a),(\forall x)\neg P(x)\vdash(\exists z)Q(z)}\,cut$$

via grade-reduction of $(\exists y)(P(y) \lor Q(y))$ we get

$$\dfrac{\dfrac{P(a) \vdash P(a)}{P(a) \vdash P(a) \lor Q(a)}\,\lor_{r_1} \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(a) \vdash P(a)}{P(a), \neg P(a) \vdash}\,\neg_l}{P(a), \neg P(a) \vdash Q(a)}\,w_r \qquad \dfrac{Q(a) \vdash Q(a)}{Q(a), \neg P(a) \vdash Q(a)}\,w_l}{P(a) \lor Q(a), \neg P(a) \vdash Q(a)}\,\lor_l}{P(a) \lor Q(a), \neg P(a) \vdash (\exists z)Q(z)}\,\exists_r}{P(a) \lor Q(a), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\,\forall_l}{P(a), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\,cut$$

via rank-reduction over $\forall_l$ we get

$$\dfrac{\dfrac{\dfrac{P(a) \vdash P(a)}{P(a) \vdash P(a) \lor Q(a)}\,\lor_{r_1} \qquad \dfrac{\dfrac{\dfrac{\dfrac{P(a) \vdash P(a)}{P(a), \neg P(a) \vdash}\,\neg_l}{P(a), \neg P(a) \vdash Q(a)}\,w_r \qquad \dfrac{Q(a) \vdash Q(a)}{Q(a), \neg P(a) \vdash Q(a)}\,w_l}{P(a) \lor Q(a), \neg P(a) \vdash Q(a)}\,\lor_l}{P(a) \lor Q(a), \neg P(a) \vdash (\exists z)Q(z)}\,\exists_r}{P(a), \neg P(a) \vdash (\exists z)Q(z)}\,cut}{P(a), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\,\forall_l$$

via rank-reduction over $\exists_r$ we get

$$\dfrac{\dfrac{\dfrac{\dfrac{P(a) \vdash P(a)}{P(a) \vdash P(a) \lor Q(a)}\,\lor_{r_1} \qquad \dfrac{\dfrac{\dfrac{P(a) \vdash P(a)}{P(a), \neg P(a) \vdash}\,\neg_l}{P(a), \neg P(a) \vdash Q(a)}\,w_r \qquad \dfrac{Q(a) \vdash Q(a)}{Q(a), \neg P(a) \vdash Q(a)}\,w_l}{P(a) \lor Q(a), \neg P(a) \vdash Q(a)}\,\lor_l}{P(a), \neg P(a) \vdash Q(a)}\,cut}{P(a), \neg P(a) \vdash (\exists z)Q(z)}\,\exists_r}{P(a), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\,\forall_l$$

via grade-reduction of $P(a) \lor Q(a)$ we get

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(a) \vdash P(a)}{P(a) \vdash P(a), Q(a)}\,w_r \qquad \dfrac{Q(a) \vdash Q(a)}{Q(a), \neg P(a) \vdash Q(a)}\,w_l}{P(a), \neg P(a) \vdash P(a), Q(a)}\,cut \qquad \dfrac{\dfrac{P(a) \vdash P(a)}{P(a), \neg P(a) \vdash}\,\neg_l}{P(a), \neg P(a) \vdash Q(a)}\,w_r}{P(a), \neg P(a), \neg P(a) \vdash Q(a), Q(a)}\,cut}{P(a), \neg P(a) \vdash Q(a), Q(a)}\,c_l}{P(a), \neg P(a) \vdash Q(a)}\,c_r}{P(a), \neg P(a) \vdash (\exists z)Q(z)}\,\exists_r}{P(a), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\,\forall_l$$

via cut-elimination over w_r we get

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(a)\vdash P(a)}{P(a)\vdash P(a),Q(a)}\,w_r}{P(a),\neg P(a)\vdash P(a),Q(a)}\,w_l \qquad \dfrac{\dfrac{P(a)\vdash P(a)}{P(a),\neg P(a)\vdash}\,\neg_l}{P(a),\neg P(a)\vdash Q(a)}\,w_r}{P(a),\neg P(a),\neg P(a)\vdash Q(a),Q(a)}\,cut}{P(a),\neg P(a)\vdash Q(a),Q(a)}\,c_l}{P(a),\neg P(a)\vdash Q(a)}\,c_r}{P(a),\neg P(a)\vdash(\exists z)Q(z)}\,\exists_r}{P(a),(\forall x)\neg P(x)\vdash(\exists z)Q(z)}\,\forall_l
$$

via rank-reduction over w_r we get

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(a)\vdash P(a)}{P(a)\vdash P(a),Q(a)}\,w_r}{P(a),\neg P(a)\vdash P(a),Q(a)}\,w_l \qquad \dfrac{P(a)\vdash P(a)}{P(a),\neg P(a)\vdash}\,\neg_l}{P(a),\neg P(a),\neg P(a)\vdash Q(a)}\,cut}{P(a),\neg P(a),\neg P(a)\vdash Q(a),Q(a)}\,w_r}{P(a),\neg P(a)\vdash Q(a),Q(a)}\,c_l}{P(a),\neg P(a)\vdash Q(a)}\,c_r}{P(a),\neg P(a)\vdash(\exists z)Q(z)}\,\exists_r}{P(a),(\forall x)\neg P(x)\vdash(\exists z)Q(z)}\,\forall_l
$$

via rank-reduction over $\neg_l$ we get

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(a)\vdash P(a)}{P(a)\vdash P(a),Q(a)}\,w_r}{P(a),\neg P(a)\vdash P(a),Q(a)}\,w_l \qquad P(a)\vdash P(a)}{P(a),\neg P(a)\vdash P(a),Q(a)}\,cut}{P(a),\neg P(a),\neg P(a)\vdash Q(a)}\,\neg_l}{P(a),\neg P(a),\neg P(a)\vdash Q(a),Q(a)}\,w_r}{P(a),\neg P(a)\vdash Q(a),Q(a)}\,c_l}{P(a),\neg P(a)\vdash Q(a)}\,c_r}{P(a),\neg P(a)\vdash(\exists z)Q(z)}\,\exists_r}{P(a),(\forall x)\neg P(x)\vdash(\exists z)Q(z)}\,\forall_l
$$

via cut-elimination over axioms we get

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(a) \vdash P(a)}{P(a) \vdash P(a), Q(a)}\,w_r}{P(a), \neg P(a) \vdash P(a), Q(a)}\,w_l}{P(a), \neg P(a), \neg P(a) \vdash Q(a)}\,\neg_l}{P(a), \neg P(a), \neg P(a) \vdash Q(a), Q(a)}\,w_r}{P(a), \neg P(a) \vdash Q(a), Q(a)}\,c_l}{P(a), \neg P(a) \vdash Q(a)}\,c_r}{P(a), \neg P(a) \vdash (\exists z)Q(z)}\,\exists_r}{P(a), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\,\forall_l$$

Now we will eliminate the cut occurring in φ_2':

$$\dfrac{\dfrac{\dfrac{Q(b) \vdash Q(b)}{Q(b) \vdash P(b) \lor Q(b)}\,\lor_{r_2}}{Q(b) \vdash (\exists y)(P(y) \lor Q(y))}\,\exists_r \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(v) \vdash P(v)}{P(v), \neg P(v) \vdash}\,\neg_l}{P(v), \neg P(v) \vdash Q(v)}\,w_r \qquad \dfrac{Q(v) \vdash Q(v)}{Q(v), \neg P(v) \vdash Q(v)}\,w_l}{P(v) \lor Q(v), \neg P(v) \vdash Q(v)}\,\lor_l}{P(v) \lor Q(v), \neg P(v) \vdash (\exists z)Q(z)}\,\exists_r}{P(v) \lor Q(v), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\,\forall_l}{(\exists y)(P(y) \lor Q(y)), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\,\exists_l}{Q(b), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\,cut$$

applying the same intermediate reduction steps as for φ_1', we get

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{Q(b) \vdash Q(b)}{Q(b) \vdash P(b), Q(b)}\,w_r \qquad \dfrac{Q(b) \vdash Q(b)}{Q(b), \neg P(b) \vdash Q(b)}\,w_l}{Q(b), \neg P(b) \vdash P(b), Q(b)}\,cut \qquad \dfrac{\dfrac{P(b) \vdash P(b)}{P(b), \neg P(b) \vdash}\,\neg_l}{P(b), \neg P(b) \vdash Q(b)}\,w_r}{Q(b), \neg P(b), \neg P(b) \vdash Q(b), Q(b)}\,cut}{Q(b), \neg P(b) \vdash Q(b), Q(b)}\,c_l}{Q(b), \neg P(b) \vdash Q(b)}\,c_r}{Q(b), \neg P(b) \vdash (\exists z)Q(z)}\,\exists_r}{Q(b), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\,\forall_l$$

via rank-reduction over w_l we get:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{Q(b) \vdash Q(b)}{Q(b) \vdash P(b), Q(b)}\,w_r \qquad Q(b) \vdash Q(b)}{Q(b) \vdash P(b), Q(b)}\,cut}{Q(b), \neg P(b) \vdash P(b), Q(b)}\,w_l \qquad \dfrac{\dfrac{P(b) \vdash P(b)}{P(b), \neg P(b) \vdash}\,\neg_l}{P(b), \neg P(b) \vdash Q(b)}\,w_r}{Q(b), \neg P(b), \neg P(b) \vdash Q(b), Q(b)}\,cut}{Q(b), \neg P(b) \vdash Q(b), Q(b)}\,c_l}{Q(b), \neg P(b) \vdash Q(b)}\,c_r}{Q(b), \neg P(b) \vdash (\exists z)Q(z)}\,\exists_r}{Q(b), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\,\forall_l$$

via cut-elimination over axioms we get:

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{Q(b) \vdash Q(b)}{Q(b) \vdash P(b), Q(b)}\, w_r}{Q(b), \neg P(b) \vdash P(b), Q(b)}\, w_l \qquad \dfrac{\dfrac{P(b) \vdash P(b)}{P(b), \neg P(b) \vdash}\, \neg_l}{P(b), \neg P(b) \vdash Q(b)}\, w_r}{Q(b), \neg P(b), \neg P(b) \vdash Q(b), Q(b)}\, cut}{Q(b), \neg P(b) \vdash Q(b), Q(b)}\, c_l}{Q(b), \neg P(b) \vdash Q(b)}\, c_r}{Q(b), \neg P(b) \vdash (\exists z)Q(z)}\, \exists_r}{Q(b), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\, \forall_l}{}
$$

applying the same intermediate reduction steps as for φ_1' we get:

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{Q(b) \vdash Q(b)}{Q(b) \vdash P(b), Q(b)}\, w_r}{Q(b), \neg P(b) \vdash P(b), Q(b)}\, w_l}{Q(b), \neg P(b), \neg P(b) \vdash Q(b)}\, \neg_l}{Q(b), \neg P(b), \neg P(b) \vdash Q(b), Q(b)}\, w_r}{Q(b), \neg P(b) \vdash Q(b), Q(b)}\, c_l}{Q(b), \neg P(b) \vdash Q(b)}\, c_r}{Q(b), \neg P(b) \vdash (\exists z)Q(z)}\, \exists_r}{Q(b), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\, \forall_l
$$

Finally we obtain the following cut-free **LK**-proof φ' with the same end-sequent as φ:

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(a) \vdash P(a)}{P(a) \vdash P(a), Q(a)}\, w_r}{P(a), \neg P(a) \vdash P(a), Q(a)}\, w_l}{P(a), \neg P(a), \neg P(a) \vdash Q(a)}\, \neg_l}{P(a), \neg P(a), \neg P(a) \vdash Q(a), Q(a)}\, w_r}{P(a), \neg P(a) \vdash Q(a), Q(a)}\, c_l}{P(a), \neg P(a) \vdash Q(a)}\, c_r}{P(a), \neg P(a) \vdash (\exists z)Q(z)}\, \exists_r}{P(a), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\, \forall_l \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{Q(b) \vdash Q(b)}{Q(b) \vdash P(b), Q(b)}\, w_r}{Q(b), \neg P(b) \vdash P(b), Q(b)}\, w_l}{Q(b), \neg P(b), \neg P(b) \vdash Q(b)}\, \neg_l}{Q(b), \neg P(b), \neg P(b) \vdash Q(b), Q(b)}\, w_r}{Q(b), \neg P(b) \vdash Q(b), Q(b)}\, c_l}{Q(b), \neg P(b) \vdash Q(b)}\, c_r}{Q(b), \neg P(b) \vdash (\exists z)Q(z)}\, \exists_r}{Q(b), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\, \forall_l
$$

$$
\dfrac{P(a), (\forall x)\neg P(x) \vdash (\exists z)Q(z) \qquad Q(b), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}{P(a) \vee Q(b), (\forall x)\neg P(x) \vdash (\exists z)Q(z)}\, \vee_l
$$

CHAPTER 4

Cut-Elimination by Resolution

We will introduce the cut-elimination method CERES (cut-elimination by resolution) which, in contrast to the reductive methods, uses the resolution principle. First we will give a motivation for the development of CERES and then give a definition of clause terms. Finally we will give a formal definition of the method CERES.

4.1 Motivation

In Chapter 3 we have seen methods which eliminate cuts by stepwise reduction of cut-complexity. These methods identify the uppermost logical operator in the cut-formula and then either eliminate it directly via grade reduction or indirectly via rank reduction. Only a small part of the proof is analysed, namely the derivation corresponding to the introduction of the uppermost logical operator [6]. The drawback of these methods is that they do not take the general structure of the proof into account. Hence, many types of redundancy in proofs are left undetected which leads to bad computational behaviour.

The method CERES was introduced by Baaz and Leitsch [3] and it analyses the global structure of an **LK**-proof φ and therefore all cut-derivations in φ are analysed simultaneously [6]. One important part of CERES is the characteristic clause set, which is extracted from an **LK**-proof and depends on the interplay between binary rules that produce ancestors of cut-formulas and those that do not [6].

It is shown that CERES can achieve a nonelementary speed-up over reductive methods [5], [6]. It is also shown in [5], that for every characteristic clause set $CL(\varphi)$ there exists a resolution refutation γ.

CERES can also be used to prove negative results about cut-elimination. For example one can show that a certain cut-free proof is not obtainable by a given one [6].

Originally CERES was developed for classical logic, but it has also been successfully extended to finitely valued logics [4], Gödel logic [2] and higher order logic [13], [22].

We will give an outline of CERES by briefly explaining its main steps. A more detailed explanation can be found in [6]. Let φ be an **LK**-derivation with end-sequent S. Then CERES consists of the following steps:

1. Skolemization of φ
   The end-sequent S is required to be Skolemized for the method CERES, i.e. there are no strong quantifier inferences operating on end-sequent ancestors. Therefore φ is Skolemized, where the eigenvariables are replaced by so-called Skolem terms. After the **LK**-proof is transformed into ACNF, the final derivation is transformed into a derivation of the original (un-Skolemized) end-sequent [5]. Note here, in our method for CERES, we do not need the Skolemization of input proofs. So this part is skipped.

2. Construction of the characteristic clause set $CL(\varphi)$
   Every instance of the cut-rule introduces two copies of a potentially new formula into φ. Then these two formulas are gradually decomposed into their atomic subformulas. Some of these atoms may end up in initial sequents of the form $C = C_i \circ C_i'$, where C_i denotes the part of C consisting of atomic cut-ancestors, and C_i' denotes the part of C consisting of ancestors of formulas occurring in the end-sequent. Starting from the initial sequents, a set of clauses $CL(\varphi)$ is constructed, which consists only of clauses composed of C_i.

3. Construction of a projection $\varphi(C_i)$ for each $C_i \in CL(\varphi)$
   Each $C_i \in CL(\varphi)$ is a subsequent of some initial sequent in φ, therefore we can obtain a cut-free derivation of a sequent $S \circ C_i$, where S is the end-sequent of φ, for every $C_i \in CL(\varphi)$. To obtain this, we skip all inferences that operate on cut-ancestors and introduce some additional weakenings in order to obtain all formulas of S, if necessary. As a consequence, the atoms of C_i remain unchanged throughout φ. The projection $\varphi(C_i)$ of C_i is then given by the derivation of the sequent $S \circ C_i$ [5].

4. Construction of a resolution refutation γ of $CL(\varphi)$
   It is shown that the characteristic clause set $CL(\varphi)$ is always unsatisfiable [6]. Hence, there always exists a resolution refutation γ of $CL(\varphi)$ by the completeness of the resolution calculus [14]. Such a refutation corresponds to the empty sequent $\vdash$ from the clauses in $CL(\varphi)$. By applying a ground projection to γ, a ground resolution refutation γ' of $CL(\varphi)$ is obtained [5].

5. Merging the projections $\varphi(C_i)$ and the ground resolution refutation γ'
   Now we need to combine the projections $\varphi(C_i)$ and the ground resolution refutation γ' of $CL(\varphi)$. This is done by applying the ground substitution σ, which defines γ', to each projection $\varphi(C_i)$ and placing $\varphi(C_i)\sigma$ immediately above the initial sequents in γ' that correspond to the same $C_i \in CL(\varphi)$. After combining the projections and γ' we obtain an **LK**-derivation of S that contains only atomic cuts, since the resolution steps in γ' can be considered as atomic cuts in **LK**. Some contractions might be necessary here, in order to obtain an **LK**-derivation of S [5].

4.2 Clause Terms

The information present in the axioms refuted by the cuts will be represented by a set of clauses. Every proof φ with cuts can be transformed into a proof φ' of the empty sequent by skipping inferences going into the end-sequent. The axioms of this refutation can be represented by clause terms [6].

The following two definitions are taken from [6] Definition 6.3.1 and Definition 6.3.2.

Definition 4.2.1. Clause term. Clause terms are $\{\oplus, \otimes\}$-terms over clause sets. More formally:

- (Finite) sets of clauses are clause terms.
- if X, Y are clause terms then $X \oplus Y$ is a clause term.
- if X, Y are clause terms then $X \otimes Y$ is a clause term.

Definition 4.2.2. Semantics of clause terms. We define a mapping $|.|$ from clause terms to sets of clauses in the following way:

$$|\mathscr{C}| = \mathscr{C} \text{ for a set of clauses } \mathscr{C}$$

$$|X \oplus Y| = |X| \cup |Y|$$

$$|X \otimes Y| = |X| \times |Y|$$

where $\mathscr{C} \times \mathscr{D} = \{C \circ D | C \in \mathscr{C}, D \in \mathscr{D}\}$.

Clause terms are equivalent if the corresponding sets of clauses are equal, i.e. $X \sim Y$ iff $|X| = |Y|$.

Definition 4.2.3. [6] Definition 6.3.3. Let σ be a substitution. We define the application of σ to clause terms as follows:

$$X\sigma = \mathscr{C}\sigma \text{ if } X = \mathscr{C} \text{ for a set of clauses } \mathscr{C}$$

$$(X \oplus Y)\sigma = X\sigma \oplus Y\sigma$$

$$(X \otimes Y)\sigma = X\sigma \otimes Y\sigma$$

Definition 4.2.4. [6] Definition 6.3.4. Let X, Y be clause terms. We define

- $X \subseteq Y$ iff $|X| \subseteq |Y|$ (i.e. iff $|X|$ is a subclause of $|Y|$
- $X \sqsubseteq Y$ iff for all $C \in |Y|$ there exists a $D \in |X|$ s.t. $D \sqsubseteq C$
- $X \leq_s Y$ iff there exists a substitution σ with $X\sigma = Y$

The operators $\oplus$ and $\otimes$ are compatible with the relations $\subset$ and $\subset$. This is formally proved in [6]. We will just state the lemmas here, for the proof we refer to [6].

Lemma 4.2.1. *[6], Lemma* 6.3.1. *Let X, Y, Z be clause terms and $X \subseteq Y$. Then*

1. $X \oplus Z \subseteq Y \oplus Z$,
2. $Z \oplus X \subseteq Z \oplus Y$,
3. $X \otimes Z \subseteq Y \otimes Z$,
4. $Z \otimes X \subseteq Z \otimes Y$.

Lemma 4.2.2. *[6], Lemma* 6.3.2. *Let X, Y, Z be clause terms and $X \sqsubseteq Y$. Then*

1. $X \oplus Z \sqsubseteq Y \oplus Z$,
2. $Z \oplus X \sqsubseteq Z \oplus Y$,
3. $X \otimes Z \sqsubseteq Y \otimes Z$,
4. $Z \otimes X \sqsubseteq Z \otimes Y$.

Replacing subterms in a clause term preserves the relations $\subseteq$ and $\sqsubseteq$. The following lemma is taken from [6] and for the proof we refer to [6].

Lemma 4.2.3. *[6] Lemma* 6.3.3. *Let λ be an occurrence in a clause term X and $Y \preceq X.\lambda$ for $\preceq \in \{\subseteq, \sqsubseteq\}$. Then $X[Y]_\lambda \preceq X$.*

4.3 The Method CERES

The method CERES is a method of cut-elimination which essentially uses the semantic information of the refutability of the cuts after the rest of the proof has been deleted.

The following two definitions are based on [6] Definition 3.1.15 and Definition 3.1.16.

Definition 4.3.1. Polarity. Let λ be an occurrence of a formula A in B.

- If $A = B$ then λ is a positive occurrence in B.
- If $B = (C \odot D)$ for $\odot \in \{\wedge, \vee\}$ and λ is a positive (negative) occurrence of A in C (or in D, respectively) then the corresponding occurrence λ' of A in B is positive (negative).
- If $B = (Qx)C$ for $Q \in \{\forall, \exists\}$ and λ is a positive (negative) occurrence of A in C then the corresponding occurrence λ' of A in B is positive (negative).
- If $B = \neg C$ and λ is a positive (negative) occurrence of A in C then the corresponding occurrence λ' of A in B is negative (positive).

If there exists a positive (negative) occurrence of a formula A in B we say that A is of positive (negative) polarity in B.

Definition 4.3.2. Strong and weak quantifiers. If $(\forall x)$ occurs positively (negatively) in B then $(\forall x)$ is called a strong (weak) quantifier. If $(\exists x)$ occurs positively (negatively) in B then $(\exists x)$ is called a weak (strong) quantifier.

Now we define Skolemization, the definition is taken from [6] Definition 6.2.1.

Definition 4.3.3. Skolemization. sk is a function which maps closed formulas into closed formulas; it is defined in the following way:

$$sk(F) = F \text{ if } F \text{ does not contain strong quantifiers.}$$

Otherwise assume that (Qx) is the first strong quantifier in F (in a tree ordering) which is in the scope of the weak quantifiers $(Q_1 x_1), ..., (Q_n x_n)$ (appearing in this order). Let f be an n-ary function symbol not occurring in F (f is a constant symbol for $n = 0$). Then $sk(F)$ is defined inductively as

$$sk(F) = sk(F_{Qy}\{y \leftarrow f(x_1, ..., x_n)\}).$$

where F_{Qy} is F after omission if (Qy). $sk(F)$ is called the (structural) Skolemization of F.

The definition of Skolemization in model theory and automated deduction mostly is dual to the definition above, i.e. in case of prenex forms the existential quantifiers are eliminated instead of the universal ones. This kind of Skolemization is called refutational Skolemization [6]. The dual kind of Skolemization (elimination of universal quantifiers) is called Herbrandization. Skolemization of sequents yields a more general framework covering both concepts [6]. The following definition is taken from [6] Definition 6.2.2.

Definition 4.3.4. Skolemization of sequents. Let S be the sequent $A_1, ..., A_n \vdash B_1, ..., B_m$ consisting of closed formulas only and

$$sk((A_1 \wedge ... \wedge A_n) \rightarrow (B_1 \vee ... \vee B_m)) = (A'_1 \wedge ... \wedge A'_n) \rightarrow (B'_1 \vee ... \vee B'_m).$$

Then the sequent

$$S' : A'_1, ..., A'_n \vdash B'_1, ..., B'_m$$

is called the Skolemization of S.

Example 4.3.1. [6] Example 6.2.1. Let S be the sequent $(\forall x)(\exists y)P(x, y) \vdash (\forall x)(\exists y)P(x, y)$. Then the Skolemization of S is $S' : (\forall x)P(x, f(x)) \vdash (\exists x)P(c, y)$ for a one-place function symbol f and a constant symbol c.

We follow [6]:
By a Skolemized proof we mean a proof of the Skolemized end sequent. Proofs with cuts can be Skolemized as well, but the cut formulas themselves cannot. Only the strong quantifiers which are ancestors of the end sequent are eliminated. Skolemization does not increase the length of proofs.

Definition 4.3.5. [6] Definition 6.2.3. Let φ be an arbitrary **LK**-proof. By $\|\varphi\|_l$ we denote the number of logical inferences and cuts in φ. Unary structural rules are not counted.

Proposition 4.3.1. *[6] Proposition* 6.2.1. *Let φ be an **LK**-proof of S from an atomic axiom set $\mathscr{A}$. Then there exists a proof $sk(\varphi)$ of $sk(S)$ (the structural Skolemization of S) from $\mathscr{A}$ s.t. $\|sk(\varphi)\|_l \leq \|\varphi\|_l$.*

Proof. See [6] Proposition 6.2.1. □

Definition 4.3.6. [6] Definition 6.2.4. Let Φ^s be the set of all **LK**-derivations with Skolemized end sequents. $\Phi^s_\emptyset$ is the set of all cut-free proofs in Φ^s and for all $i \geq 0$ Φ^s_i is the set of all proofs in Φ^s with cut-complexity $\leq i$.

Hence, on Skolemized proofs cut-elimination means to transform a derivation in Φ^s into a derivation in Φ^s_0 [6].

The following two definitions are based on [6] Definition 6.4.1 and 6.4.2.

Definition 4.3.7. Characteristic term. Consider an **LK**-derivation φ of S and let Ω be the set of all occurrences of cut formulas in φ. Then the characteristic (clause) term $\Phi(\varphi)$ is defined inductively via $\Phi(\varphi)/\nu$ for occurrences of sequents ν in φ: Let ν be the occurrence of an initial sequent in φ. Then $\Phi(\varphi)/\nu = \{S(\nu, \Omega)\}$ (see Definition 3.2.14 in [6]).

Now assume that all the clause terms $\Phi(\varphi)/\nu$ are constructed for sequent occurrences ν in φ with $depth(\nu) \leq k$. Now consider an occurrence ν with $depth(\nu) = k + 1$. We distinguish the following cases:

1. ν is the conclusion of μ, i.e. a unary rule applied to μ gives ν. Here we simply define $\Phi(\varphi)/\nu = \Phi(\varphi)/\mu$.

2. ν is the conclusion of μ_1 and μ_2, i.e. a binary rule ξ applied to μ_1 and μ_2 gives ν.

   a) The occurrences of the auxiliary formulas of ξ are ancestors of Ω, thus the formulas occur on $S(\mu_1, \Omega), S(\mu_2, \Omega)$. Then $\Phi(\varphi)/\nu = \Phi(\varphi)/\mu_1 \oplus \Phi(\varphi)/\mu_2$.

   b) The occurrences of the auxiliary formulas of ξ are not ancestors of Ω. Here we define $\Phi(\varphi)/\nu = \Phi(\varphi)/\mu_1 \otimes \Phi(\varphi)/\mu_2$.

In binary inferences, either the occurrences of both auxiliary formulas are ancestors of Ω or none of them.

Finally the characteristic term $\Phi(\varphi)$ is defined as $\Phi(\varphi)/\nu$ where ν is the end-sequent occurrence.

Definition 4.3.8. Characteristic clause set. Let φ be an **LK**-derivation and $\Phi(\varphi)$ be the characteristic term of φ. Then $CL(\varphi) = |\Phi(\varphi)|$ is called the characteristic clause set of φ.

It is shown in [6] that the set of characteristic clauses $CL(\varphi)$ for a proof φ is unsatisfiable.

Proposition 4.3.2. *[6] Proposition* 6.4.1. *Let φ be regular **LK**-proof of a closed sequent and $\varphi \in \Phi^s$. Then $CL(\varphi)$ is unsatisfiable.*

Proof. See [6] Proposition 6.4.1. □

We follow [6]:
Let $\varphi \in \Phi^s$ be a deduction of $S : \Gamma \vdash \Delta$ and $CL(\varphi)$ be the characteristic clause set of φ. Then $CL(\varphi)$ is unsatisfiable and by the completeness of resolution there exists a resolution refutation γ of $CL(\varphi)$ [14]. By applying a ground projection to γ we obtain a ground resolution refutation of $CL(\varphi)$, we call it γ'. γ' is also an **LK**-deduction of $\vdash$ from ground instances of $CL(\varphi)$. Hence γ' may serve a s a skeleton of an Φ_0^s-proof ϕ of $\Gamma \vdash \Delta$ itself. To construct ϕ from γ' we need projections that replace φ by cut-free deductions $\varphi(C)$ of $\overline{P}, \Gamma \vdash \Delta, \overline{Q}$ for clauses $C : \overline{P} \vdash \overline{Q}$ in $CL(\varphi)$. Roughly speaking, the projections of the proof φ are obtained by skipping all the inferences leading to a cut. Note that we obtain a characteristic clause set in the end sequent as a residue. Thus a projection is a cut-free derivation of the end sequent S + some atomic formulas. For the application of projections we need to have Skolemized end-sequents, otherwise eigenvariable conditions could be violated.

The following definition is taken from [6].

Definition 4.3.9. [6] Definition 6.4.4. Let φ be an **LK**-proof, ν a node in φ and Ω a set of formula occurrences in φ. Then we define $\overline{S}(\nu, \Omega)$ by

$$Seq(\nu) = S(\nu, \Omega) \circ \overline{S}(\nu, \Omega).$$

$\overline{S}(\nu, \Omega)$ is the subsequent of $Seq(\nu)$ consisting of the non-ancestors of Ω, i.e. of the ancestors of the end-sequent.

Lemma 4.3.3. *[6] Lemma* 6.4.1. *Let φ be a deduction in Φ^s of a sequent S from an axiom set $\mathscr{A}$ and let C be a clause in $CL(\varphi)$. Then there exists a deduction $\varphi[C] \in \Phi_\emptyset^s$ of $C \circ S$ from $\mathscr{A}$ and*

$$\|\varphi[C]\|_l \leq \|\varphi\|_l$$

Proof. See [6] Lemma 6.4.1. □

The following definition is taken from [6].

Definition 4.3.10. Projection. Let $\varphi \in \Phi^s$ be a proof and $C \in CL(\varphi)$. Then the **LK**-proof $\varphi[C]$ is called the projection of φ w.r.t. C. Let σ be an arbitrary substitution, then $\varphi[C\sigma]$ is defined as $\varphi[C]\sigma$ and is also called the projection of φ w.r.t. $C\sigma$, i.e. instances of projections are also projections.

Definition 4.3.11. [6] Definition 6.4.6. Let φ be a proof of a closed sequent S and $\varphi \in \Phi^s$. We define a set

$$PES(\varphi) = \{S \circ C\sigma \mid C \in CL(\varphi), \sigma \text{ a substitution}\}$$

which contains all end sequents of projections w.r.t. φ.

Now we state the main result of this section.

Theorem 4.3.4. *[6] Theorem* 6.4.1. *CERES is a cut-elimination method, i.e. for every proof φ of a sequent S in Φ^s CERES produces a proof ϕ of S s.t. $\phi \in \Phi^s_0$.*

Proof. See [6] Theorem 6.4.1. □

The context product allows to extend a proof on all nodes by a clause. The definition is taken from [6] Definition 4.6.3.

Definition 4.3.12. Context product. Let C be a sequent and φ be an **LK**-derivation such that no free variable in C occurs as eigenvariable in φ. We define the left context product $C \star \varphi$ of C and φ (which gives a proof of $C \circ S$) inductively:

- If φ consists only of the root node ν and $Seq(\nu) = S$ then $C \star \varphi$ is a proof consisting only of a node μ such that $Seq(\mu) = C \circ S$.
- Assume that φ is of the form

$$\cfrac{\begin{matrix}(\varphi') \\ S'\end{matrix}}{S}\ \xi$$

  where ξ is a unary rule. Assume also that $C \star \varphi'$ is already defined and is an **LK**-derivation of $C \circ S'$. Then we define $C \star \varphi$ as:

$$\cfrac{\begin{matrix}(C \star \varphi') \\ C \circ S'\end{matrix}}{C \circ S}\ \xi$$

  $C \star \varphi$ is well defined also for the rules $\forall_r$ and $\forall_l$ as C does not contain free variables which are eigenvariables in φ.
- Assume that φ is of the form

$$\cfrac{\begin{matrix}(\varphi_1) \\ S_1\end{matrix} \qquad \begin{matrix}(\varphi_2) \\ S_2\end{matrix}}{S}\ \xi$$

  and $C \star \varphi_1$ is a proof of $C \circ S_1$, $C \star \varphi_2$ is a proof of $C \circ S_2$. Then we define the proof $C \star \varphi$ as

$$\frac{\frac{\dfrac{(C \star \varphi_1)}{C \circ S_1} \qquad \dfrac{(C \star \varphi_2)}{C \circ S_2}}{S'}\ \xi}{C \circ S}\ s^*$$

Note that, if ξ is the cut rule, restoring the context after application of ξ might require weakening; otherwise s^* stands for applications of contractions.

The right context product $\varphi \star C$ is defined in the same way.

The proof constructed by the CERES-method is a specific type of proof in Φ^s which contains parts of the original proof in the form of projections and this type of proof is called a CERES normal form. The following definition is based on [6] Definition 6.4.7.

Definition 4.3.13. CERES normal form. Let $\varphi \in \Phi^s$ be a proof of S and γ a ground resolution refutation of the unsatisfiable set of clauses $CL(\varphi)$. γ is also a proof in Φ^s from $CL(\varphi)$. First $\gamma' : S \star \gamma$ is constructed, where γ' is an **LK**-derivation of S from $PES(\varphi)$. Then define $\varphi(\gamma')$ by replacing all initial clauses $C \circ S$ in γ' by the projections $\varphi[C]$. By definition $\varphi(\gamma')$ is an **LK**-proof of S in Φ_0^s and is called the CERES-normal form of φ w.r.t. γ.

CHAPTER 5

CERES in Higher-Order Logic

The CERES-method was originally defined as a cut-elimination method for first-order logic, where it is not quite clear how to handle induction. It is represented by an axiom scheme and first-order logic does not have any object-level tools to handle schemes. Since induction can be represented by a single axiom in second-order logic, it became of major interest to move CERES to higher-order logic. This was established in [13] and [22] where CERES$^\omega$ was introduced. In first-order logic, proof Skolemization is used since the CERES-method performs a transformation which is not sound in the presence of eigenvariables [13]. Proof Skolemization removes inferences which obey eigenvariable conditions, hence this transformation can be performed. In higher-order logic, proof Skolemization is not compatible with the quantifier rules. As a solution to this problem a new calculus, $\mathbf{LK}_{sk}$, is defined in [13] and [22], where eigenvariables are replaced by Skolem terms. The proof projections are then proofs that may be locally unsound, due to the violations of eigenvariable conditions, but they fulfil some global soundness properties. The transformation into an **LK**-proof is possible [13] and [22] which proves soundness of $\mathbf{LK}_{sk}$.

5.1 Types, languages and Skolem terms

For CERES$^\omega$ a version of Church's simple theory of types is used [7].

Definition 5.1.1. Base types. Base types are defined as the set $BT = \{\iota, o\}$ of the types of individuals ι and booleans o.

The following definition is taken from [22] Definition 2.1.1.

Definition 5.1.2. Types. We define the set $\mathscr{T}$ of types along with their order o inductively:

1. $BT \subseteq \mathscr{T}$, for all $t \in BT$ $o(t) = 1$.

2. If $T_1, t_2 \in \mathscr{T}$ then $t = t_1 \to t_2 \in \mathscr{T}$ and $o(t) = max(o(t_1), o(t_2)) + 1$.

Note that $\to$ associates to the right. The following definition is taken from [22] Definition 2.1.9.

Definition 5.1.3. Skolem symbols. Let $\alpha, \beta_1, ..., \beta_n$ be types. Then the list $\sigma = \beta_1, ..., \beta_n, \alpha$ is called a signature (for a Skolem symbol). For each signature σ let $\sigma^T = \beta_1 \to ... \to \beta_n \to \alpha$ and let $\mathscr{K}_\sigma \subseteq \mathscr{C}_{\sigma^T}$ be a denumerable set of constant symbols of type σ^T s.t. if σ_1 and σ_2 are different signatures, then $\mathscr{K}_{\sigma_1}$ and $\mathscr{K}_{\sigma_2}$ are disjoint. $C \in \mathscr{K}_\sigma$ is called a Skolem symbol of signature σ with arity n. Then define the Herbrand Universe as the set of all $T \in \mathscr{E}$ (where $\mathscr{E}$ denotes the Herbrand Universe, i.e. only expressions contained in the Herbrand Universe) s.t. whenever a Skolem symbol of arity n has an occurrence in T it is applied to at least n arguments. Furthermore, if a variable has a free occurrence in any of these arguments, that occurrence is also free in T.

5.2 The calculus $\mathbf{LK}_{sk}$

The following definition is taken from [13] Definition 2.

Definition 5.2.1. Labelled Sequents. A label is a finite multiset of terms. A labelled sequent is a sequent $F_1, ..., F_n \vdash F_{n+1}, ..., F_m$ together with labels l_i for $1 \leq i \leq m$ and we write $\langle F_1 \rangle^{l_1}, ..., \langle F_n \rangle^{l_n} \vdash \langle F_{n+1} \rangle^{l_{n+1}}, ..., \langle F_m \rangle^{l_m}$. We identify labelled formulas with empty labels with the respective unlabelled formulas. If S is a labelled sequent, then the reduct of S is S where all labels are empty. If $\mathscr{C}$ is a set of labelled sequents, then the reduct of $\mathscr{C}$ is $\{S \mid S \text{ is a reduct of some } S' \in \mathscr{C}\}$.

Substitution is extended to labelled sequents [13]. Let σ be a substitution and

$$S = \langle F_1 \rangle^{l_1}, ..., \langle F_n \rangle^{l_n} \vdash \langle F_{n+1} \rangle^{l_{n+1}}, ..., \langle F_m \rangle^{l_m}$$

then

$$S\sigma = \langle F_1\sigma \rangle^{l_1\sigma}, ..., \langle F_n\sigma \rangle^{l_n\sigma} \vdash \langle F_{n+1}\sigma \rangle^{l_{n+1}\sigma}, ..., \langle F_m\sigma \rangle^{l_m\sigma}$$

The purpose of the labels is that they will track quantifier instantiation information throughout prooftrees and that they enable to combine resolution refutations and sequent calculus proofs in a certain way [13]. The following definition is taken from [13] Definition 3.

Definition 5.2.2. $\mathbf{LK}_{sk}$ rules.
Labelled quantifier rules:

$$\frac{\Gamma \vdash \Delta, \langle F(fS_1...S_n) \rangle^l}{\Gamma \vdash \Delta, \langle \forall_\alpha F \rangle^l} \forall_r^{sk}$$

where $l = S_1, ..., S_n$ and if $\tau(S_i) = \alpha_i$ for $1 \leq i \leq n$ then $f \in \mathscr{K}_{\alpha_1,...,\alpha_n,\alpha}$ is a Skolem symbol. An application of this rule is called source inference of $fS_1...S_m$ and $fS_1...S_m$ is called the Skolem term of this inference. Note that we do not impose an eigenvariable or eigenterm restriction on this rule.

$$\frac{\langle FT \rangle^{l,T}, \Gamma \vdash \Delta}{\langle \forall_\alpha F \rangle^{l}, \Gamma \vdash \Delta} \, \forall_l^{sk}$$

T is called the substitution term of this inference. The $\exists_l^{sk}$ and $\exists_r^{sk}$ rules are defined analogously. The $\forall_r^{sk}$ and $\exists_l^{sk}$ rules will be called strong labelled quantifier rules and the $\forall_l^{sk}$ and $\exists_r^{sk}$ will be called weak labelled quantifier rules. The other rules of **LK** are transferred directly to $\mathbf{LK}_{sk}$:
Propositional rules:

$$\frac{\langle F \rangle^l, \Gamma \vdash \Delta \qquad \langle G \rangle^l, \Pi \vdash \Lambda}{\langle F \vee G \rangle^l, \Gamma, \Pi \vdash \Delta, \Lambda} \, \vee_l$$

$$\frac{\Gamma \vdash \Delta, \langle F \rangle^l}{\Gamma \vdash \Delta, \langle F \vee G \rangle^l} \, \vee_{r_1}$$

The rest of the propositional rules of **LK** are adapted analogously.
Structural rules:

$$\frac{\Gamma \vdash \Delta, \langle F \rangle^l, \langle F \rangle^l}{\Gamma \vdash \Delta, \langle F \rangle^l} \, c_r$$

$$\frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, \langle F \rangle^l} \, w_r$$

and analogously for c_l and w_l. An $\mathbf{LK}_{sk}$-tree is a tree formed according to the rules of $\mathbf{LK}_{sk}$, such that all leaves are of the form $\langle F \rangle^{l_1} \vdash \langle F \rangle^{l_2}$ for some formula F and some labels l_1, l_2. The axiom partner of $\langle F \rangle^{l_1}$ is defined to be $\langle F \rangle^{l_2}$, and vice versa. Let π be an $\mathbf{LK}_{sk}$-tree with end-sequent S. If S does not contain Skolem terms or free variables, and all labels in S are empty, then S is called proper. If the end-sequent of π is proper, we say that π is proper. Note that $\mathbf{LK}_{sk}$ is a cut free calculus.

Example 5.2.1. [13], Example 1. The following figure shows a proper $\mathbf{LK}_{sk}$-tree of a valid sequent:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\langle S(f(\lambda x.\neg S(x)))\rangle^{\lambda x.\neg S(x)} \vdash \langle S(f(\lambda x.\neg S(x)))\rangle^{\lambda x.\neg S(x)}}{\langle \neg S(f(\lambda x.\neg S(x)))\rangle^{\lambda x.\neg S(x)}, \langle S(f(\lambda x.\neg S(x)))\rangle^{\lambda x.\neg S(x)} \vdash} \neg_l}{\langle S(f(\lambda x.\neg S(x)))\rangle^{\lambda x.\neg S(x)} \vdash \langle \neg\neg S(f(\lambda x.\neg S(x)))\rangle^{\lambda x.\neg S(x)}} \neg_r}{\vdash \langle S(f(\lambda x.\neg S(x))) \to \neg\neg S(f(\lambda x.\neg S(x)))\rangle^{\lambda x.\neg S(x)}} \to_r}{\vdash \langle (\forall z)(S(z) \to \neg\neg S(z))\rangle^{\lambda x.\neg S(x)}} \forall_r^{sk}}{\vdash \langle (\exists Y)(\forall z)(S(z) \to \neg Y(z))\rangle} \exists_r^{sk}}{\vdash \langle (\forall X)(\exists Y)(\forall z)(X(z) \to \neg Y(z))\rangle} \forall_r^{sk}$$

where $S \in \mathscr{K}_{\iota \to o}$, $f \in \mathscr{K}_{\iota \to o, \iota}$, and the substitution term of the $\exists_r^{sk}$ is $\lambda x.\neg S(x)$. Note that although the labels in the axiom coincide, this is not required in general.

$\mathbf{LK}_{sk}$-trees are unsound without some restrictions. Consider the following example:

Example 5.2.2. [13], Example 2. Consider the following $\mathbf{LK}_{sk}$-tree of $(\exists x)P(x) \vdash (\forall x)P(x)$:

$$\dfrac{\dfrac{P(s) \vdash P(s)}{(\exists x)P(x) \vdash P(s)}\ \exists_l^{sk}}{(\exists x)P(x) \vdash (\forall x)P(x)}\ \forall_r^{sk}$$

where $s \in \mathscr{K}_\iota$. The reason of unsoundness in this example is that we are allowed to use the same Skolem term for distinct and unrelated strong quantifier inferences in $\mathbf{LK}_{sk}$-trees. Note that there are no labels in the example above.

In the following some definitions and facts about occurrences in $\mathbf{LK}_{sk}$-trees will be introduced.

Proposition 5.2.1. *[13] Proposition* 4. *Let ω be a formula occurrence in a proper $\boldsymbol{LK}_{sk}$-tree π with label $\{T_1, ..., T_n\}$. Then $T_1, ..., T_n$ are exactly the substitution terms of the weak labelled quantifier inferences operating on descendants of ω.*

Proof. By induction in the number of sequents between ω and the end-sequent of π. For details see [13] Proposition 4. □

Definition 5.2.3. Paths. Let μ be a sequence of formula occurrences $\mu_1, ..., \mu_n$ in an $\mathbf{LK}_{sk}$-tree. If it holds that for all $1 \leq i \leq n$ μ_i is an immediate ancestor (immediate descendant) of μ_{i+1}, then μ is called a downwards (upwards) path. If μ is a downwards (upwards) path ending in an occurrence in the end-sequent (a leaf), then μ is called maximal.

Definition 5.2.4. Homomorphic paths. If ω is a formula occurrence, then we denote the formula at ω by $F(\omega)$. If μ is a sequence of formula occurrences, we define $F(\mu)$ to be μ where every formula occurrence ω is replaced by $F(\omega)$ and repetitions are omitted. Two sequences of formula occurrences μ, ν are called homomorphic, if $F(\mu) = F(\nu)$.

Proposition 5.2.2. *[13] Proposition* 5. *Let π be a proper $\boldsymbol{LK}_{sk}$-tree and ρ be a strong labelled quantifier inference in π with Skolem term S and auxiliary formula α. Let μ be a maximal downwards path starting at α. Then $FV(S) = FV(\mu)$, where $FV(S)$ denotes the set of free variables in S.*

Proof. As π is proper, its end-sequent does not contain free variables. Hence all free variables in μ are contained in substitution terms of weak labelled quantifier inferences, and they are exactly the free variables of S by [13] Proposition 4. □

Proposition 5.2.3. *[13] Proposition* 6. *Let α_1, α_2 be formula occurrences. If there exists a downwards path from α_1 to α_2, then it is unique.*

Proof. Every formula occurrence has at most one direct descendant. □

Corollary. *[13] Corollary* 1. *If α is a formula occurrence, then there exists a unique maximal downwards path starting at α.*

The investigation of paths allows to define a relation between inferences in a tree that are connected in a strong sense through paths. The following definition is based on [13] Definition 6.

Definition 5.2.5. Homomorphic Inferences. Let α_1 and α_2 be formula occurrences in an $\mathbf{LK}_{sk}$-tree π. Let c be a contraction inference below α_1 and α_2 with auxiliary occurrences γ_1 and γ_2. Then α_1 and α_2 are homomorphic in c if the downwards paths $\alpha_1, ..., \gamma_1$ and $\alpha_2, ..., \gamma_2$ exist and are homomorphic. α_1 and α_2 are homomorphic if there exists a c s.t. they are homomorphic in c.

Let ρ_1 and ρ_2 be inferences of the same type and let α_1^1 (α_1^2) and α_2^1 (α_2^2) be their auxiliary formula occurrences. Then ρ_1 and ρ_2 are called homomorphic if there exists a contraction inference c s.t. α_1^1 and α_2^1 are homomorphic in c and α_1^{12} and α_2^2 are homomorphic in c. Call this contraction inference the uniting contraction of ρ_1 and ρ_2.

Example 5.2.3. [13], Example 4. Consider the following $\mathbf{LK}_{sk}$-tree π:

$$\dfrac{\dfrac{\dfrac{\dfrac{\langle P(s)\rangle^s \vdash P(s)}{(\forall x)P(x) \vdash P(s)}\ \forall_l^{sk}}{(\forall x)P(x) \vdash (\forall x)P(x)}\ \forall_r^{sk}(1) \qquad \dfrac{\dfrac{\langle P(s)\rangle^s \vdash P(s)}{\langle P(s)\rangle^s \vdash (\forall x)P(x)}\ \forall_r^{sk}(3)}{(\forall x)P(x) \vdash (\forall x)P(x)}\ \forall_l^{sk}}{(\forall x)P(x) \lor (\forall x)P(x) \vdash (\forall x)P(x), (\forall x)P(x)}\ \lor_l}{(\forall x)P(x) \lor (\forall x)P(x) \vdash (\forall x)P(x)}\ c_r(2)$$

The inferences (1) and (3) in π are homomorphic and (2) is their uniting contraction. More concretely, let μ be the path from the auxiliary formula of (1) to the auxiliary formula of (2). Let ν be the path from the auxiliary formula of (3) to the auxiliary formula of (2). Then $F(\mu) = P(s), (\forall x)P(x) = F(\nu)$.

Now consider π':

$$\dfrac{\dfrac{\dfrac{\dfrac{\langle P(s_1)\rangle^{s_1} \vdash P(s_1)}{(\forall x)P(x) \vdash P(s_1)}\ \forall_l^{sk}}{(\forall x)P(x) \vdash (\forall x)P(x)}\ \forall_r^{sk}(1) \qquad \dfrac{\dfrac{\langle P(s_2)\rangle^{s_2} \vdash P(s_2)}{\langle P(s_2)\rangle^{s_2} \vdash (\forall x)P(x)}\ \forall_r^{sk}(3)}{(\forall x)P(x) \vdash (\forall x)P(x)}\ \forall_l^{sk}}{(\forall x)P(x) \lor (\forall x)P(x) \vdash (\forall x)P(x), (\forall x)P(x)}\ \lor_l}{(\forall x)P(x) \lor (\forall x)P(x) \vdash (\forall x)P(x)}\ c_r(2)$$

In π' there are no homomorphic inferences. The reason is that the auxiliary formulas of the $\forall_r^{sk}$ applications differ. Define μ and ν as above, then $F(\mu) = P(s_1), (\forall x)P(x)$ and $F(\nu) = P(s_2), (\forall x)P(x)$, hence $F(\mu) \neq F(\nu)$.

Proposition 5.2.4. *[13] Proposition* 7. *If two strong labelled quantifier inferences are homomorphic, they have identical Skolem terms.*

Proof. See [13] Proposition 7. □

Now we can define the notion of an $\mathbf{LK}_{sk}$-proof. The following definition is based on [13] Definition 7.

Definition 5.2.6. Weak regularity and $\mathbf{LK}_{sk}$-proofs. Let π be an $\mathbf{LK}_{sk}$-tree with end-sequent S. π is weakly regular if for all distinct strong labelled quantifier inferences ρ_1 and ρ_2 in π it holds that if ρ_1 and ρ_2 have identical Skolem terms, then ρ_1 and ρ_2 are homomorphic. π is an $\mathbf{LK}_{sk}$-proof if it is weakly regular and proper.

Particularly the $\mathbf{LK}_{sk}$-tree in Example 5.2 is not weakly regular. Note that in ordinary **LK**, it follows directly from the definition of regularity that all strong quantifier inferences in a regular **LK**-tree π fulfil the eigenvariable condition and hence, that **LK**-trees are **LK**-proofs. Here, inferences are allowed to use the same eingenterm, provided they are homomorphic.

Since $\mathbf{LK}_{sk}$ is cut-free, ordinary **LK** is connected with the rules of $\mathbf{LK}_{sk}$. The following definition is taken from [13] Definition 8.

Definition 5.2.7. $\mathbf{LK}_{skc}$-trees. An $\mathbf{LK}_{skc}$-tree is a tree formed according to the rules of $\mathbf{LK}_{sk}$ and **LK** s.t.

1. rules of **LK** operate only on cut-ancestors, and

2. rules of $\mathbf{LK}_{sk}$ operate only on end-sequent ancestors.

Hence the cut-ancestors in an $\mathbf{LK}_{skc}$-tree have empty labels.

5.3 The resolution calculus $\mathscr{R}_{al}$

Since CERES$^\omega$ is a CERES-method for higher-order logic, it needs a different resolution calculus than the one defined for the ordinary CERES-method. In particular, in [13] and [22] the resolution calculus $\mathscr{R}_{al}$ is introduced. It is quite close to Andrews' resolution calculus $\mathscr{R}$ [1], since this one can be regarded as the most simple formulation of a resolution calculus for higher-order logic.

We will not give a list of the rules and deductions of this resolution calculus here, instead we refer to [13] and [22] and just state a definition and the main result as a theorem.

Definition 5.3.1. Relative completeness of $\mathscr{R}_{al}$. Let $\mathscr{S}$ be a set of labelled sequents. $\mathscr{R}_{al}$ is relatively complete if the following holds: If there exists an $\mathscr{R}$-refutation of the reduct of $\mathscr{S}$, then there exists an $\mathscr{R}_{al}$-refutation of $\mathscr{S}$.

Theorem 5.3.1. *[1] Theorem* 5.3*. Let $\mathscr{S}$ be a set of sentences. If there exists a τ-refutation of $\mathscr{S}$ then there exists an $\mathscr{R}$-refutation of $\mathscr{S}$ (where τ is the system defined in [1]).*

5.4 Cut-elimination for $\mathbf{LK}_{sk}$

LK-proofs can be translated to $\mathbf{LK}_{sk}$-proofs. The notions of paths, homomorphic inferences and weak regularity are extended to $\mathbf{LK}_{skc}$-trees. Let π be an $\mathbf{LK}_{skc}$-tree with end-sequent S. Then π is called an $\mathbf{LK}_{skc}$-proof if π is weakly regular and proper. The next definition is based on [13] Definition 10.

Definition 5.4.1. $\mathbf{LK}_{skc}$ regularity. Let π be an $\mathbf{LK}_{skc}$-tree. π is called regular if

1. each strong labelled quantifier inference has a unique Skolem symbol and
2. the eigenvariable of each strong quantifier inference ρ only occurs above ρ in π.

Proposition 5.4.1. *[13] Proposition* 9*. Let π be an $\mathbf{LK}_{skc}$-tree. If π is regular, then π is weakly regular.*

Lemma 5.4.2. *[13] Lemma* 1*, Skolemization. Let π be a regular **LK**-proof of S. Then there exists a regular $\mathbf{LK}_{skc}$-proof ϕ of S.*

Proof. By induction on the height of ρ, where ρ is an inference in π with conclusion $F_1, ..., F_n \vdash F_{n+1}, ..., F_m$. For a detailed proof we refer to [13] Lemma 1. □

We follow [13]:
Let π be an $\mathbf{LK}_{skc}$-tree and let S be a sequent in π. Then $cutanc(S)$ is the subsequent of S consisting of the cut-ancestors of S and $esanc(S)$ is the subsequent of S consisting of the end-sequent ancestors of S. For any sequent $S = cutanc(S) \circ esanc(S)$. Let ρ be a unary inference, σ a binary inference, ϕ_1 and ϕ_2 $\mathbf{LK}_{sk}$-trees, then $\rho(\phi_1)$ is the $\mathbf{LK}_{sk}$-tree obtained by applying ρ to the end-sequent of φ_1, and $\sigma(\phi_1, \phi_2)$ is the $\mathbf{LK}_{sk}$-tree obtained from the $\mathbf{LK}_{sk}$-trees ϕ_1 and ϕ_2 by applying σ. Let P, Q be sets of $\mathbf{LK}_{sk}$-trees. Then $P^{\Gamma \vdash \Delta} = \{\phi^{\Gamma \vdash \Delta} | \phi \in P\}$, where $\phi^{\Gamma \vdash \Delta}$ is ϕ followed by weakenings adding $\Gamma \vdash \Delta$, and $P \times_\sigma Q = \{\sigma(\phi_1, \phi_2) | \phi_1 \in P, \phi_2 \in Q\}$. The following definition is taken from [13] Definition 11.

Definition 5.4.2. Characteristic Sequent Set and Projections. Let π be a regular $\mathbf{LK}_{skc}$-proof. For each inference ρ in π, we define a set of $\mathbf{LK}_{sk}$-trees, the set of projections $\mathscr{P}_\rho(\pi)$, and a set of labelled sequents, the characteristic sequent set $CS_\rho(\pi)$.

- If ρ is an axiom with conclusion $S = \langle A \rangle^{l_1} \vdash \langle A \rangle^{l_2}$, distinguish:
  - $cutanc(S) = S$, then $CS_\rho(\pi) = \mathscr{P}_\rho(\pi) = \emptyset$.
  - $cutanc(S) \neq S$, distinguish:
    * If $cutanc(S) = \vdash \langle A \rangle^{l_2}$ then $CS_\rho(\pi) = \{\vdash \langle A \rangle^{l_1}\}$ and $\mathscr{P}_\rho(\pi) = \{\langle A \rangle^{l_1} \vdash \langle A \rangle^{l_1}\}$,
    * if $cutanc(S) = \langle A \rangle^{l_1} \vdash$ then $CS_\rho(\pi) = \{\langle A \rangle^{l_2} \vdash\}$ and $\mathscr{P}_\rho(\pi) = \{\langle A \rangle^{l_2} \vdash \langle A \rangle^{l_2}\}$,
    * if $cutanc(S) = \vdash$ then $CS_\rho(\pi) = \{\vdash\}$ and $\mathscr{P}_\rho(\pi) = \{S\}$.

- If ρ is a unary inference with immediate predecessor ρ' with $\mathscr{P}_\rho(\pi) = \phi_1, ..., \phi_n$, distinguish:
  - ρ operates in ancestors of cut-formulas, then
  $$\mathscr{P}_\rho(\pi) = \mathscr{P}_{\rho'}(\pi)$$
  - ρ operates in ancestors of the end-sequent, then
  $$\mathscr{P}_\rho(\pi) = \{\rho(\phi_1), ..., \rho(\phi_n)\}$$

  In any case, $CS_\rho(\pi) = CS_{\rho'}(\pi)$.

- Let ρ be a binary inference with immediate predecessors ρ_1 and ρ_2.
  - If ρ operates on ancestors of cut-formulas, let $\Gamma_i \vdash \Delta_i$ be the ancestors of the end-sequent in the conclusion sequent of ρ_i and define
  $$\mathscr{P}_\rho(\pi) = \mathscr{P}_{\rho_1}(\pi)^{\Gamma_2 \vdash \Delta_2} \cup \mathscr{P}_{\rho_2}(\pi)^{\Gamma_1 \vdash \Delta_1}$$
  For the characteristic sequent set, define
  $$CS_\rho(\pi) = CS_{\rho_1}(\pi) \cup CS_{\rho_2}(\pi)$$
  - if ρ operates on ancestors of the end-sequent, then
  $$\mathscr{P}_\rho(\pi) = \mathscr{P}_{\rho_1}(\pi) \times_\rho \mathscr{P}_{\rho_2}(\pi)$$
  For the characteristic sequent set, define
  $$CS_\rho(\pi) = CS_{\rho_1}(\pi) \times CS_{\rho_2}(\pi)$$

The set of projections of π, $\mathscr{P}(\pi)$ is defined as $\mathscr{P}_{\rho_0}(\pi)$, and the characteristic sequent set of π, $CS(\pi)$ is defined as $CS_{\rho_0}(\pi)$, where ρ_0 is the last inference in π.

Note that for $\mathbf{LK}_{skc}$-proofs π containing only atomic axioms, $CS(\pi)$ consists of sequents containing only atomic formulas [13].

Proposition 5.4.3. *[13] Proposition* 10. *Let π be a regular $\boldsymbol{LK}_{skc}$-proof. Then there exists an **LK**-refutation of the reduct of $CS(\pi)$.*

Proof. For each inference ρ with conclusion S in π, an **LK**-tree γ_ρ of the reduct of $cutanc(S)$ from the reduct of $CS_\rho(\pi)$ is defined inductively. Then for the last inference ρ in π, γ_ρ is the desired **LK**-refutation. For a detailed proof we refer to [13]. □

The next definition is based on [13] Definition 12.

Definition 5.4.3. Restrictedness. Let $\mathscr{S}$ be a set of formula occurrences in an $\mathbf{LK}_{skc}$-tree π. π is $\mathscr{S}$-linear if no inferences operate on ancestors of occurrences in $\mathscr{S}$. If no inferences except contraction operate on ancestors of occurrences in $\mathscr{S}$, then π is $\mathscr{S}$-restricted. If $\mathscr{S}$ is the set of occurrences of cut-formulas of π and π is $\mathscr{S}$-restricted, we say that π is restricted.

Proposition 5.4.4. *[13] Proposition* 11. *Let π be an $\boldsymbol{LK}_{skc}$-tree and $\mathscr{S}$ a set of formula occurrences in π that is closed under descendants, and let π be $\mathscr{S}$-linear. If π' is obtained from π by replacing all labels of ancestors of occurrences in $\mathscr{S}$ by the empty label, then π' is an $\boldsymbol{LK}_{skc}$-tree.*

Proof. As π is $\mathscr{S}$-linear, no inferences operates on the respective occurrences. No inference has restrictions on labels of context formulas, except that direct descendants have the same label as their direct ancestors, an no axioms pose restrictions on labels, therefore the proposition holds. □

The next definition is based on [13] Definition 13.

Definition 5.4.4. Skolem Parallel. Let π_1 and π_2 be $\mathbf{LK}_{skc}$-trees and let ρ_1 and ρ_2 be strong labelled quantifier inferences in π_1 and π_2 with Skolem terms S_1 and S_2, respectively. ρ_1 and ρ_2 are Skolem parallel if for all substitutions σ_1 and σ_2 it holds that if $S_1\sigma_1 = S_2\sigma_2$ then $\mu_1\sigma_1$ and $\mu_2\sigma_2$ are homomorphic, where μ_1 and μ_2 are the maximal downwards paths starting at S_1 and S_2, respectively. π_1 and π_2 are Skolem parallel if for all strong labelled quantifier inferences ρ_1, ρ_2 in π_1, π_2 respectively, ρ_1 and ρ_2 are Skolem parallel.

With this definition, we are able to state the following proposition.

Proposition 5.4.5. *[13] Proposition* 12. *Let π_1 and π_2 be $\boldsymbol{LK}_{skc}$-trees and σ a substitution. If π_1, π_2 are Skolem parallel, then $\pi_1\sigma, \pi_2\sigma$ are.*

Proof. See [13] Proposition 12. □

The following definitions are taken from [13] Definition 14 and Definition 15.

Definition 5.4.5. Axiom Labels. Let π be an $\mathbf{LK}_{skc}$-tree and let ω be a formula occurrence in π. Let μ be an ancestor of ω that occurs in an axiom A. Then A is called a source axiom for ω. Let $\mathscr{S}$ be a set of formula occurrences in π. π has suitable axiom labels w.r.t. $\mathscr{S}$ if for all formula occurrences ω in $\mathscr{S}$, the source axioms of ω are of the form $\langle F\rangle^l \vdash \langle F\rangle^l$.

Definition 5.4.6. Balancedness. Let π be an $\mathbf{LK}_{skc}$-tree and let $\mathscr{S}$ be a set of formula occurrences in π. π is called $\mathscr{S}$-balanced if for every axiom $\langle F\rangle^{l_1} \vdash \langle F\rangle^{l_2}$ in π at least one occurrence of F is an ancestor of a formula occurrence in $\mathscr{S}$. π is called balanced if π is $\mathscr{S}$-balanced where $\mathscr{S}$ is the set of end-sequent occurrences of π.

Now the projections can be defined. The following definition is based on [13] Definition 16.

Definition 5.4.7. CERES-projections. Let S be a proper sequent and C be a sequent. An $\mathbf{LK}_{skc}$-tree π is called a CERES-projection for (S, C) if the end-sequent of π is $S \circ C$ and π is weakly regular, $\mathscr{O}_C$-linear, $\mathscr{O}_S$-balanced, restricted and has suitable axiom labels w.r.t. $\mathscr{O}_C$, where $\mathscr{O}_C$ and $\mathscr{O}_S$ are the set of formula occurrences of C and S in the end-sequent of π, respectively.

Let $\mathscr{C}$ be a set of sequents. A set of $\mathbf{LK}_{skc}$-trees $\mathscr{P}$ is called a set of CERES-projections for $(S, \mathscr{C})$ if for all $C \in \mathscr{C}$ there exists a $\pi(C) \in \mathscr{P}$ s.t. $\pi(C)$ is a CERES-projection for (S, C) and for all $\pi_1, \pi_2 \in \mathscr{P}$, π_1 and π_2 are Skolem parallel.

Lemma 5.4.6. *[13] Lemma 2. Let π be a regular $\mathbf{LK}_{skc}$-proof of S. Then $\mathscr{P}(\pi)$ is a set of CERES-projections for $(S, CS(\pi))$. Furthermore, for all $\phi \in \mathscr{P}(\pi)$, $|\phi| \leq |\pi|$.*

Proof. See [13] Lemma 2. □

It is shown in [13] Lemma 3 that if we have a set of sequents $\mathscr{C}$ and a proper sequent S, together with a CERES-projection for $(S, \mathscr{C})$, if there exists an $\mathscr{R}_{al}$-refutation of $\mathscr{C}$, there exists a restricted, weakly regular, balanced $\mathbf{LK}_{skc}$-tree of S. The following lemma is taken from [13] Lemma 4.

Lemma 5.4.7. *Let π be a restricted $\mathbf{LK}_{skc}$-proof of S. Then there exists an $\mathbf{LK}_{sk}$-proof of S.*

Proof. By induction on the number of cut inferences in π. For a detailed proof we refer to [13] Lemma 4. □

Finally one of the main theorems of CERES$^{\omega}$ can be stated.

Theorem 5.4.8. *[13] Theorem 2. Let π be a regular, proper $\mathbf{LK}_{skc}$-proof of S s.t. there exists an $\mathscr{R}_{al}$-refutation of $CS(\pi)$. Then there exists an $\mathbf{LK}_{sk}$-proof of S.*

Proof. By [13] Lemma 2 and Lemma 3 there exists a restricted $\mathbf{LK}_{skc}$-proof of S. By Lemma 5.4.7 there exists an $\mathbf{LK}_{sk}$-proof of S. □

Since CERES$^{\omega}$ is a cut-elimination method for **LK**, it can be shown that $\mathbf{LK}_{sk}$-proofs can be translated to cut-free **LK**-proofs.

5.5 Soundness of $\mathbf{LK}_{sk}$

It is shown in [13] and [22] that weak regularity suffices for soundness of $\mathbf{LK}_{sk}$-proofs. To show the soundness of $\mathbf{LK}_{sk}$, $\mathbf{LK}_{sk}$-proofs are transformed into **LK**-proofs. This is accomplished by permuting inferences and substituting eigenvariables for Skolem terms. Since in $\mathbf{LK}_{sk}$-proofs it may be the case that two strong labelled inferences in a common branch use the same Skolem term, a kind of redundancy may be present [13]. This prevents an eigenterm condition form holding and therefore we cannot substitute an eigenvariable for the Skolem term. This redundancy can be eliminated by sequential pruning. The following definition is based on [13] Definition 18.

Definition 5.5.1. Sequential pruning. Consider an $\mathbf{LK}_{sk}$-tree π and let ρ and ρ' be inferences in π. If ρ and ρ' are on a common branch in π, they are called sequential. The set of sequential homomorphic pairs is defined as follows:

$$SHP(\pi) = \{\langle \rho, \rho' \rangle \mid \rho, \rho' \text{ homomorphic in } \pi \text{ and } \rho, \rho' \text{ sequential}\}$$

π is sequentially pruned if $SHP(\pi) = \emptyset$.

For pruning sequential homomorphic pairs, we will analyse the permutation of contraction inferences over independent inferences. Independent inferences are defined as follows (the definition is based on [13] Definition 19):

Definition 5.5.2. Independent inferences. Let ρ and σ be two inferences s.t. ρ is above σ. Then ρ and σ are independent if the auxiliary formula of σ is not a descendant of the main formula of ρ.

The following definition is taken from [13] Definition 20.

Definition 5.5.3. The relation $\triangleright_c$. We will now define the rewrite relation $\triangleright_c$ for $\mathbf{LK}_{sk}$-trees π and π', where we assume the inferences c_* and σ to be independent:

1. If π is

$$\cfrac{\cfrac{\Pi, \Pi, \Gamma \vdash \Delta, \Lambda, \Lambda}{\Pi, \Gamma \vdash \Delta, \Lambda}\, c_*}{\Pi, \Gamma' \vdash \Delta', \Lambda}\, \sigma$$

and π' is

$$\cfrac{\cfrac{\Pi, \Pi, \Gamma \vdash \Delta, \Lambda, \Lambda}{\Pi, \Pi, \Gamma' \vdash \Delta', \Lambda, \Lambda}\, \sigma}{\Pi, \Gamma' \vdash \Delta', \Lambda}\, c_*$$

then $\pi \triangleright_c^1 \pi'$.

2. If π is

$$\cfrac{\cfrac{\Pi, \Pi, \Gamma \vdash \Delta, \Lambda, \Lambda}{\Pi, \Gamma \vdash \Delta, \Lambda}\, c_* \qquad \Sigma \vdash \Theta}{\Pi, \Gamma' \vdash \Delta', \Lambda}\, \sigma$$

and π' is

$$\cfrac{\cfrac{\Pi, \Pi, \Gamma \vdash \Delta, \Lambda, \Lambda \qquad \Sigma \vdash \Theta}{\Pi, \Pi, \Gamma' \vdash \Delta', \Lambda, \Lambda}\, \sigma}{\Pi, \Gamma' \vdash \Delta', \Lambda}\, c_*$$

then $\pi \rhd_c^1 \pi'$.

3. If π is

$$\dfrac{\Sigma \vdash \Theta \qquad \dfrac{\Pi, \Pi, \Gamma \vdash \Delta, \Lambda, \Lambda}{\Pi, \Gamma \vdash \Delta, \Lambda}\, c_*}{\Pi, \Gamma' \vdash \Delta', \Lambda}\, \sigma$$

and π' is

$$\dfrac{\dfrac{\Sigma \vdash \Theta \qquad \Pi, \Pi, \Gamma \vdash \Delta, \Lambda, \Lambda}{\Pi, \Pi, \Gamma' \vdash \Delta', \Lambda, \Lambda}\, \sigma}{\Pi, \Gamma' \vdash \Delta', \Lambda}\, c_*$$

then $\pi \rhd_c^1 \pi'$.

The $\rhd_c$ relation is then defined as the transitive and reflexive closure of the compatible closure of the $\rhd_c^1$ relation.

Lemma 5.5.1. *[13] Lemma 7. Let π be a weakly regular $\boldsymbol{LK}_{sk}$-tree of S. If $\pi \rhd_c \phi$ then ϕ is a weakly regular $\boldsymbol{LK}_{sk}$-tree of S.*

Proof. By induction on the length of the $\rhd_c$-rewrite sequence. The case of $\pi = \phi$ is trivial, so assume there exists a subtree φ of π s.t. $\varphi \rhd_c^1 \varphi'$ and ϕ is obtained from π by replacing φ by φ'. Then the end-sequent of ϕ is the same as that of π. Weak regularity is preserved, the paths in ϕ and π are the same modulo some repetitions. □

Lemma 5.5.2. *[13] Lemma 8. Let π be an $\boldsymbol{LK}_{sk}$-tree with end-sequent S s.t. π is not sequentially pruned. Then there exists an $\boldsymbol{LK}_{sk}$-tree π' of the same end-sequent s.t.*

$$|SHP(\pi')| < |SHP(\pi)|.$$

Furthermore, if π is weakly regular, then so is π'.

Proof. See [13] Lemma 8. □

Hence we are able to state the following lemma:

Lemma 5.5.3. *[13] Lemma 9, Sequential pruning. Let π be an $\boldsymbol{LK}_{sk}$-tree of S, then there exists an $\boldsymbol{LK}_{sk}$-tree π' of S s.t. π' is sequentially pruned. Furthermore, if π is weakly regular, then so is π'.*

Proof. Repeated application of the Lemma 5.16. □

Now we need to show that $\mathbf{LK}_{sk}$-proofs can be translated into **LK**-proofs. The proof will be effective and based on permuting inferences and pruning. First we need a notational convention.

$$\Gamma, A^1 = \Gamma, A$$

$$\Gamma, A^0 = \Gamma$$

and let $i, i_1, ..., i_4 \in \{0, 1\}$, $\overline{x} = |x - 1|$. In the following transformations, the labels of the labelled formula occurrences will not be displayed. The reason is that we always leave them unchanged. The following definition is taken from [22] Definition 4.1.29.

Definition 5.5.4. The relation $\rhd_u$ This relation is used to permute down a unary logical inference ρ over an inference σ, assuming that ρ and σ are independent. We do not write down the cases involving $\wedge_r$, $\rightarrow_l$, $\rightarrow_r$ inferences, since they are analogous. In case 1, σ is a unary logical inference, in case 2, σ is a weakening inference, in case 3, σ is a contraction inference and in cases $4-5$, σ is an $\vee_l$ inference. We define a relation $\rhd_u^1$ between $\mathbf{LK}_{sk}$-trees π and π':

1. If π is

$$\cfrac{\cfrac{F^{i_1}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, F^{\overline{i_1}}}{M^{i_3}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, M^{\overline{i_3}}}\;\rho}{M^{i_3}, N^{i_4}, \Gamma \vdash \Delta, N^{\overline{i_4}}, M^{\overline{i_3}}}\;\sigma$$

   and π' is

$$\cfrac{\cfrac{F^{i_1}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, F^{\overline{i_1}}}{F^{i_1}, N^{i_4}, \Gamma \vdash \Delta, N^{\overline{i_4}}, F^{\overline{i_1}}}\;\sigma}{M^{i_3}, N^{i_4}, \Gamma \vdash \Delta, N^{\overline{i_4}}, M^{\overline{i_3}}}\;\rho$$

   then $\pi \rhd_u^1 \pi'$.

2. If π is

$$\cfrac{\cfrac{F^{i_1}, \Gamma \vdash \Delta, F^{\overline{i_1}}}{M^{i_2}, \Gamma \vdash \Delta, M^{\overline{i_2}}}\;\rho}{N^{i_3}, M^{i_2}, \Gamma \vdash \Delta, M^{\overline{i_2}}, N^{\overline{i_3}}}\;\sigma$$

   and π' is

$$\cfrac{\cfrac{F^{i_1}, \Gamma \vdash \Delta, F^{\overline{i_1}}}{N^{i_3}, F^{i_1}, \Gamma \vdash \Delta, F^{\overline{i_1}}, N^{\overline{i_3}}}\;\sigma}{N^{i_3}, M^{i_2}, \Gamma \vdash \Delta, M^{\overline{i_2}}, N^{\overline{i_3}}}\;\rho$$

then $\pi \rhd^1_u \pi'$.

3. If π is

$$\dfrac{\dfrac{F^{i_1}, G^{i_2}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, G^{\overline{i_2}}, F^{\overline{i_1}}}{M^{i_3}, G^{i_2}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, G^{\overline{i_2}}, M^{\overline{i_3}}}\ \rho}{M^{i_3}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, M^{\overline{i_3}}}\ \sigma$$

and π' is

$$\dfrac{\dfrac{F^{i_1}, G^{i_2}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, G^{\overline{i_2}}, F^{\overline{i_1}}}{F^{i_1}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, F^{\overline{i_1}}}\ \sigma}{M^{i_3}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, M^{\overline{i_3}}}\ \rho$$

then $\pi \rhd^1_u \pi'$.

4. If π is

$$\dfrac{\dfrac{F^{i_1}, G_1, \Gamma \vdash \Delta, F^{\overline{i_1}}}{M^{i_2}, G_1, \Gamma \vdash \Delta, M^{\overline{i_2}}}\ \rho \qquad G_2, \Pi \vdash \Lambda}{G_1 \vee G_2, M^{i_2}, \Gamma, \Pi \vdash \Delta, \Lambda, M^{\overline{i_2}}}\ \sigma$$

and π' is

$$\dfrac{\dfrac{F^{i_1}, G_1, \Gamma \vdash \Delta, F^{\overline{i_1}} \qquad G_2, \Pi \vdash \Lambda}{G_1 \vee G_2, F^{i_1}, \Gamma, \Pi \vdash \Delta, \Lambda, F^{\overline{i_1}}}\ \sigma}{G_1 \vee G_2, M^{i_2}, \Gamma, \Pi \vdash \Delta, \Lambda, M^{\overline{i_2}}}\ \rho$$

then $\pi \rhd^1_u \pi'$.

5. If π is

$$\dfrac{G_1, \Gamma \vdash \Delta \qquad \dfrac{F^{i_1}, G_2, \Pi \vdash \Lambda, F^{\overline{i_1}}}{M^{i_2}, G_2, \Pi \vdash \Lambda, M^{\overline{i_2}}}\ \rho}{G_1 \vee G_2, M^{i_2}, \Gamma, \Pi \vdash \Delta, \Lambda, M^{\overline{i_2}}}\ \sigma$$

and π' is

$$\dfrac{\dfrac{G_1, \Gamma \vdash \Delta \qquad F^{i_1}, G_2, \Pi \vdash \Lambda, F^{\overline{i_1}}}{G_1 \vee G_2, F^{i_1}, \Gamma, \Pi \vdash \Delta, \Lambda, F^{\overline{i_1}}}\ \sigma}{G_1 \vee G_2, M^{i_2}, \Gamma, \Pi \vdash \Delta, \Lambda, M^{\overline{i_2}}}\ \rho$$

then $\pi \rhd_u^1 \pi'$.

Finally, we define the $\rhd_u$ relation as the transitive and reflexive closure of the compatible closure of the $\rhd_u^1$ relation.

Lemma 5.5.4. *[13] Lemma 10. Let π be a weakly regular $\boldsymbol{LK}_{sk}$-tree of S. If $\pi \rhd_u \phi$ then ϕ is a weakly regular $\boldsymbol{LK}_{sk}$-tree of S.*

Proof. By induction on the length of the $\rhd_u$-rewrite sequence. The case of $\pi = \phi$ is trivial, so assume there exists a subtree φ of π s.t. $\varphi \rhd_u^1 \varphi'$ and ϕ is obtained from π by replacing φ by φ'. Then the end-sequent of ϕ is the same as that of π. Weak regularity is preserved since the paths in ϕ and π are the same modulo some repetitions. □

The following definition is based on [22] Definition 4.1.31 and [13] Definition 22.

Definition 5.5.5. The relation $\rhd_b$ This relation is used to permute down a $\vee_l$ inference ρ (the cases for $\wedge_r$, $\rightarrow_l$ are analogous) together with some contractions. In the proof trees, the indicated occurrences of F_1 and F_2 will be the auxiliary occurrences of ρ. We will now define the rewrite relation $\rhd_b$ on $\mathbf{LK}_{sk}$-tree, where we assume ρ and σ to be independent. Cases $1-3$ treat the case of σ being a unary logical inference, in case 4 σ is a weakening inference, in cases $5-6$ σ is a contraction inference and in cases $7-9$ σ is $\vee_l$.

1. π is

$$\cfrac{\cfrac{\cfrac{F_1, \Pi, \Gamma_1, G^{i_1} \vdash \Delta_1, G^{\overline{i_1}}, \Lambda \qquad F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda}{F_1 \vee F_2, \Pi, \Pi, \Gamma_1, \Gamma_2, G^{i_1} \vdash \Delta_1, G^{\overline{i_1}}, \Delta_2, \Lambda, \Lambda}\,\rho}{G^{i_1}, F_1 \vee F_2, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, G^{\overline{i_1}}}\,c_*}{M^{i_2}, F_1 \vee F_2, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, M^{\overline{i_2}}}\,\sigma$$

and π' is

$$\cfrac{\cfrac{\cfrac{G^{i_1}, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda, G^{\overline{i_1}}}{M^{i_2}, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda, M^{\overline{i_2}}}\,\sigma \qquad F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda}{F_1 \vee F_2, M^{i_2}, \Pi, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda, M^{\overline{i_2}}}\,\rho}{F_1 \vee F_2, M^{i_2}, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, M^{\overline{i_2}}}\,c_*$$

then $\pi \rhd_b^1 \pi'$.

2. π is

$$\cfrac{\cfrac{\cfrac{F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda \qquad F_2, \Pi, \Gamma_2, G^{i_1} \vdash \Delta_2, \Lambda, G^{\overline{i_1}}}{F_1 \vee F_2, \Pi, \Pi, \Gamma_1, \Gamma_2, G^{i_1} \vdash \Delta_1, \Delta_2, \Lambda, \Lambda, G^{\overline{i_1}}}\,\rho}{G^{i_1}, F_1 \vee F_2, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, G^{\overline{i_1}}}\,c_*}{M^{i_2}, F_1 \vee F_2, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, M^{\overline{i_2}}}\,\sigma$$

and π' is

$$\dfrac{\dfrac{F_1,\Pi,\Gamma_1 \vdash \Delta_1,\Lambda \qquad \dfrac{G^{i_1},F_2,\Pi,\Gamma_2 \vdash \Delta_2,\Lambda,G^{\overline{i_1}}}{M^{i_2},F_2,\Pi,\Gamma_2 \vdash \Delta_2,\Lambda,M^{\overline{i_2}}}\,\sigma}{F_1 \vee F_2, M^{i_2},\Pi,\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,\Lambda,M^{\overline{i_2}}}\,\rho}{F_1 \vee F_2, M^{i_2},\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,M^{\overline{i_2}}}\,c_*$$

then $\pi \rhd_b^1 \pi'$.

3. π is

$$\dfrac{\dfrac{\dfrac{F_1,\Pi,G^{i_1},\Gamma_1 \vdash \Delta_1,\Lambda,G^{\overline{i_1}} \qquad F_2,\Pi,G^{i_1},\Gamma_2 \vdash \Delta_2,\Lambda,G^{\overline{i_1}}}{F_1 \vee F_2,\Pi,G^{i_1},\Pi,G^{i_1},\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,G^{\overline{i_1}},\Lambda,G^{\overline{i_1}}}\,\rho}{G^{i_1},F_1 \vee F_2,\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,G^{\overline{i_1}}}\,c_*}{M^{i_2},F_1 \vee F_2,\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,M^{\overline{i_2}}}\,\sigma$$

and π' is

$$\dfrac{\dfrac{\dfrac{G^{i_1},F_1,\Pi,\Gamma_1 \vdash \Delta_1,\Lambda,G^{\overline{i_1}}}{M^{i_2},F_1,\Pi,\Gamma_1 \vdash \Delta_1,\Lambda,M^{\overline{i_2}}}\,\sigma \qquad \dfrac{G^{i_1},F_2,\Pi,\Gamma_2 \vdash \Delta_2,\Lambda,G^{\overline{i_1}}}{M^{i_2},F_2,\Pi,\Gamma_2 \vdash \Delta_2,\Lambda,M^{\overline{i_2}}}\,\sigma}{F_1 \vee F_2,\Pi,M^{i_2},\Pi,M^{i_2},\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,M^{\overline{i_2}},\Lambda,M^{\overline{i_2}}}\,\rho}{F_1 \vee F_2,\Pi,M^{i_2},\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,M^{\overline{i_2}}}\,c_*$$

then $\pi \rhd_b^1 \pi'$.

4. π is

$$\dfrac{\dfrac{\dfrac{F_1,\Pi,\Gamma_1 \vdash \Delta_1,\Lambda \qquad F_2,\Pi,\Gamma_2 \vdash \Delta_2,\Lambda}{F_1 \vee F_2,\Pi,\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,\Lambda}\,\rho}{F_1 \vee F_2,\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda}\,c_*}{M^{i},F_1 \vee F_2,\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,M^{\overline{i}}}\,w_*$$

and π' is

$$\dfrac{\dfrac{\dfrac{F_1,\Pi,\Gamma_1 \vdash \Delta_1,\Lambda}{M^{i},F_1,\Pi,\Gamma_1 \vdash \Delta_1,\Lambda,M^{\overline{i}}}\,w_* \qquad F_2,\Pi,\Gamma_2 \vdash \Delta_2,\Lambda}{F_1 \vee F_2,M^{i},\Pi,\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,\Lambda,M^{\overline{i}}}\,\rho}{F_1 \vee F_2,M^{i},\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,M^{\overline{i}}}\,c_*$$

then $\pi \rhd_b^1 \pi'$.

5. π is

$$\dfrac{\dfrac{\dfrac{F_1,\Pi,\Gamma_1,G^i,G^i \vdash \Delta_1,\Lambda,G^{\bar{i}},G^{\bar{i}} \qquad F_2,\Pi,\Gamma_2 \vdash \Delta_2,\Lambda}{F_1 \vee F_2,\Pi,\Pi,\Gamma_1,\Gamma_2,G^i,G^i \vdash \Delta_1,\Delta_2,\Lambda,\Lambda,G^{\bar{i}},G^{\bar{i}}}\,\rho}{G^i,G^i,F_1 \vee F_2,\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,G^{\bar{i}},G^{\bar{i}}}\,c_*}{G^i,F_1 \vee F_2,\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,G^{\bar{i}}}\,\sigma$$

and π' is

$$\dfrac{\dfrac{\dfrac{F_1,\Pi,\Gamma_1,G^i,G^i \vdash \Delta_1,\Lambda,G^{\bar{i}},G^{\bar{i}}}{F_1,\Pi,\Gamma_1,G^i \vdash \Delta_1,\Lambda,G^{\bar{i}}}\,\sigma \qquad F_2,\Pi,\Gamma_2 \vdash \Delta_2,\Lambda}{F_1 \vee F_2,\Pi,\Pi,\Gamma_1,\Gamma_2,G^i \vdash \Delta_1,\Delta_2,\Lambda,\Lambda,G^{\bar{i}}}\,\rho}{G^i,F_1 \vee F_2,\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,G^{\bar{i}}}\,c_*$$

then $\pi \rhd_b^1 \pi'$.

6. π is

$$\dfrac{\dfrac{\dfrac{F_1,\Pi,\Gamma_1 \vdash \Delta_1,\Lambda \qquad F_2,\Pi,\Gamma_2,G^i,G^i \vdash \Delta_2,\Lambda,G^{\bar{i}},G^{\bar{i}}}{F_1 \vee F_2,\Pi,\Pi,\Gamma_1,\Gamma_2,G^i,G^i \vdash \Delta_1,\Delta_2,\Lambda,\Lambda,G^{\bar{i}},G^{\bar{i}}}\,\rho}{G^i,G^i,F_1 \vee F_2,\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,G^{\bar{i}},G^{\bar{i}}}\,c_*}{G^i,F_1 \vee F_2,\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,G^{\bar{i}}}\,\sigma$$

and π' is

$$\dfrac{\dfrac{F_1,\Pi,\Gamma_1 \vdash \Delta_1,\Lambda \qquad \dfrac{F_2,\Pi,\Gamma_2,G^i,G^i \vdash \Delta_2,\Lambda,G^{\bar{i}},G^{\bar{i}}}{F_2,\Pi,\Gamma_2,G^i \vdash \Delta_2,\Lambda,G^{\bar{i}}}\,\sigma}{F_1 \vee F_2,\Pi,\Pi,\Gamma_1,\Gamma_2,G^i \vdash \Delta_1,\Delta_2,\Lambda,\Lambda,G^{\bar{i}}}\,\rho}{G^i,F_1 \vee F_2,\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,G^{\bar{i}}}\,c_*$$

then $\pi \rhd_b^1 \pi'$.

7. π is

$$\dfrac{\dfrac{\dfrac{F_1,\Pi,\Gamma_1,G_1 \vdash \Delta_1,\Lambda \qquad F_2,\Pi,\Gamma_2 \vdash \Delta_2,\Lambda}{F_1 \vee F_2,\Pi,\Pi,\Gamma_1,G_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda,\Lambda}\,\rho}{G_1,F_1 \vee F_2,\Pi,\Gamma_1,\Gamma_2 \vdash \Delta_1,\Delta_2,\Lambda}\,c_* \qquad G_2,\Sigma \vdash \Theta}{G_1 \vee G_2,F_1 \vee F_2,\Pi,\Gamma_1,\Gamma_2,\Sigma \vdash \Theta,\Delta_1,\Delta_2,\Lambda}\,\sigma$$

and π' is

$$\dfrac{\dfrac{\dfrac{G_1, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda \qquad G_2, \Sigma \vdash \Theta}{G_1 \lor G_2, F_1, \Pi, \Gamma_1, \Sigma \vdash \Theta, \Delta_1, \Lambda}\,\sigma \qquad F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda}{F_1 \lor F_2, G_1 \lor G_2, \Pi, \Pi, \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda, \Lambda}\,\rho}{F_1 \lor F_2, G_1 \lor G_2, \Pi, \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda}\,c_*$$

then $\pi \rhd_b^1 \pi'$.

8. π is

$$\dfrac{\dfrac{\dfrac{F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda \qquad F_2, \Pi, \Gamma_2, G_1 \vdash \Delta_2, \Lambda}{F_1 \lor F_2, \Pi, \Pi, \Gamma_1, \Gamma_2, G_1 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda}\,\rho}{G_1, F_1 \lor F_2, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda}\,c_* \qquad G_2, \Sigma \vdash \Theta}{G_1 \lor G_2, F_1 \lor F_2, \Pi, \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda}\,\sigma$$

and π' is

$$\dfrac{\dfrac{F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda \qquad \dfrac{G_1, F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda \qquad G_2, \Sigma \vdash \Theta}{G_1 \lor G_2, F_2, \Pi, \Gamma_2, \Sigma \vdash \Theta, \Delta_2, \Lambda}\,\sigma}{F_1 \lor F_2, G_1 \lor G_2, \Pi, \Pi, \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda, \Lambda}\,\rho}{F_1 \lor F_2, G_1 \lor G_2, \Pi, \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda}\,c_*$$

then $\pi \rhd_b^1 \pi'$.

9. π is

$$\dfrac{\dfrac{\dfrac{F_1, \Pi, G_1, \Gamma_1 \vdash \Delta_1, \Lambda \qquad F_2, \Pi, , G_1\Gamma_2 \vdash \Delta_2, \Lambda}{F_1 \lor F_2, \Pi, G_1, \Pi, G_1, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda}\,\rho}{F_1 \lor F_2, \Pi, G_1, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda}\,c_* \qquad G_2, \Sigma \vdash \Theta}{G_1 \lor G_2, F_1 \lor F_2, \Pi, \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda}\,\sigma$$

and π' is

$$\dfrac{\dfrac{\dfrac{G_1, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda \qquad G_2, \Sigma \vdash \Theta}{G_1 \lor G_2, F_1, \Pi, \Gamma_1, \Sigma \vdash \Theta, \Delta_1, \Lambda}\,\sigma \qquad (\phi)}{F_1 \lor F_2, \Pi, G_1 \lor G_2, \Pi, G_1 \lor G_2, \Gamma_1, \Gamma_2, \Sigma, \Sigma \vdash \Theta, \Theta, \Delta_1, \Delta_2, \Lambda, \Lambda}\,\rho}{F_1 \lor F_2, G_1 \lor G_2, \Pi, \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda}\,c_*$$

where ϕ is

$$\dfrac{G_1, F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda \qquad G_2, \Sigma \vdash \Theta}{G_1 \lor G_2, F_2, \Pi, \Gamma_2, \Sigma \vdash \Theta, \Delta_2, \Lambda}\,\sigma$$

then $\pi \rhd_b^1 \pi'$.

The $\rhd_b$ relation is defined as the transitive and reflexive closure of the compatible closure of the $\rhd_b^1$ relation.

Note that in the transformations above the contraction c_* contracts Π, Π to Π, but this is just one special case. The application of c_* could also mean that Π, Π are left unchanged. In particular, c_* contracts Π, Π to Π, Π' where Π' can be either Π or empty.

Lemma 5.5.5. *[13] Lemma 11. Let π be a weakly regular $\boldsymbol{LK}_{sk}$-tree of S. If $\pi \rhd_b \phi$ then ϕ is a weakly regular $\boldsymbol{LK}_{sk}$-tree of S.*

Proof. By induction on the length of the $\rhd_b$-rewrite sequence. For details we refer to [13] Lemma 11. □

Now we will introduce some definitions that are used in the the translation of an $\mathbf{LK}_{sk}$-proof into an $\mathbf{LK}_{sk}$-proof which fulfils an eigenterm condition. The definitions are taken from [13].

Definition 5.5.6. Parallel trees. Let π be an $\mathbf{LK}_{sk}$-tree and let ξ be a branch in π. Let σ and ρ be inferences on ξ and w.l.o.g. let σ be above ρ. Let $\xi_1, ..., \xi_n$ be the binary inferences between σ and ρ. For $1 \leq i \leq n$ let λ_i be the subproofs ending in a premise sequent of ξ_i s.t. λ_i do not contain σ. Then $\lambda_1, ..., \lambda_n$ are called the parallel trees between σ and ρ.

Definition 5.5.7. Blocking and correctly placed inferences. Let σ be a strong labelled quantifier inference in π with Skolem term S and let ρ be a weak labelled quantifier inference in π with substitution term T. ρ blocks σ if ρ is below σ and T contains S. σ is correctly placed if no weak labelled quantifier inference in π blocks σ.

Quantifier inferences in an $\mathbf{LK}_{sk}$-proof π will be rearranged in such a way that there are no eigenterm violations. Hence, it is possible to convert the $\mathbf{LK}_{sk}$-proof into an $\mathbf{LK}$-proof [13].

Lemma 5.5.6. *[13] Lemma 13. Let π be an $\boldsymbol{LK}_{sk}$-proof of S. Then there exists an $\boldsymbol{LK}_{sk}$-proof π' of S s.t. all strong labelled quantifier inferences in π' are correctly placed.*

Proof. For the proof of this lemma we refer to [13] Lemma 13. □

Finally, we are able to state one main result of this section. We will just state the theorem and refer for the proof to [13] Theorem 3.

Theorem 5.5.7. *Soundness. Let π be an $\boldsymbol{LK}_{sk}$-proof of S. Then there exists a cut-free $\boldsymbol{LK}$-proof of S.*

The main theorem on CERES$^\omega$ can now be stated, again we just state the Theorem and refer for the proof to [13] Theorem 4.

Theorem 5.5.8. *[13] Theorem 4. Let π be a regular, proper $\boldsymbol{LK}_{skc}$-proof of S s.t. there exists an $\mathscr{R}_{al}$-refutation of $CS(\pi)$. Then there exists a cut-free $\boldsymbol{LK}$-proof of S.*

Completeness of $\mathscr{R}_{al}$ implies completeness of the cut-elimination method [13]:

Theorem 5.5.9. *[13] Theorem 5. Assume completeness of $\mathscr{R}_{al}$. Let π be an **LK**-proof of a proper sequent S. Then there exists a cut-free **LK**-proof of S.*

Proof. We refer to [13] Theorem 5. □

Note: $\mathscr{R}_{al}$ is complete for CERES^1!

Theorem 5.5.10. *Completeness of $\mathscr{R}_{al}$. $\mathscr{R}_{al}$ is complete for CERES^1.*

Proof. The source of potential incompleteness of CERES^ω (problem still unsolved) lies in the "dynamic" Skolemization of higher-order resolution. □

CHAPTER 6

Complexity Analysis of CERES

It is shown in [6] that cut-elimination is intrinsically nonelementary. This means that also CERES, applied to Statman's sequence, produces a nonelementary blowup w.r.t. the size of the input proof. In this section we will briefly explain the main source of complexity in CERES and state which CERES parts behave nonelementary on a worst-case sequence. First we need some definitions, which are taken from [6] Definition 4.2.8 and Definition 4.3.1.

Definition 6.0.8. Proof complexity. Let S be an arbitrary sequent and $\mathscr{A}$ be an axiom set. We define

$$PC^{\mathscr{A}}(S) = min\{\|\varphi\| \mid \varphi \in \Phi^{\mathscr{A}} \text{ and } \varphi \text{ proves } S\}.$$

$PC^{\mathscr{A}}(S)$ is called the proof complexity of S w.r.t. $\mathscr{A}$.

Definition 6.0.9. [6] Definition 4.3.1. Let $e : \mathbb{N}^2 \to \mathbb{N}$ be the following function

$$e(0, m) = m$$

$$e(n + 1, m) = 2^{e(n,m)}.$$

A function $f : \mathbb{N}^k \to \mathbb{N}^m$ for $k, m \geq 1$ is called elementary if there exists an $n \in \mathbb{N}$ and a Turing machine T computing f s.t. the computing time of T on input $(l_1, ..., l_k)$ is less than or equal to $e(n, |(l_1, ..., l_k)|)$ where $|\ |$ denotes the maximum norm on $\mathbb{N}^k$.

The function $s : \mathbb{N} \to \mathbb{N}$ is defined as $s(n) = e(n, 1)$ for $n \in \mathbb{N}$. A function which is not elementary is called nonelementary.

It can be shown that the size of the characteristic clause set is exponential in the size of the input proof. We will state the lemma here and for the proof we refer to [6] Lemma 6.5.1.

Lemma 6.0.11. *[6] Lemma* 6.5.1. *Let t be a clause term, then*

$$\||t|\| \leq 2^{\|t\|}$$

(the symbolic size of the set of clauses defined by a clause term is at most exponential in that of the term).

Proposition 6.0.12. *[6] Proposition* 6.5.1. *For every $\varphi \in \Phi^s$ $|CL(\varphi)| \leq 2^{\|\varphi\|}$.*

Proof. $CL(\varphi) = \Theta(\varphi)$. So, by the previously stated lemma,

$$|CL(\varphi)| \leq 2^{\|\Theta(\varphi)\|}.$$

Obviously $\|\Theta(\varphi)\| \leq \|\varphi\|$ and therefore

$$|CL(\varphi)| \leq 2^{\|\varphi\|}.$$

□

The length of the resolution refutation γ of $CL(\varphi)$ has a bigger impact on the source of complexity in the CERES-method [6].

Lemma 6.0.13. *[6] Lemma* 6.5.2. *Let γ be a resolution refutation of a set of clauses $\mathscr{C}$. Then there exists a ground resolution refutation γ' of $\mathscr{C}$ with the following properties:*

- $l(\gamma) = l(\gamma')$,
- $\|\gamma\| \leq \|\gamma'\|$,
- $\|\gamma'\| \leq 5 * l(\gamma)^2 * \|\mathscr{C}\| * 2^{5*l(\gamma)^2*\|\mathscr{C}\|}$.

Proof. See [6] Lemma 6.5.2. □

We can show that any sequence of resolution refutations of the characteristic clause sets of the Statman sequence $(\gamma_n)_{n\in\mathbb{N}}$ is of nonelementary size w.r.t. the proof complexity of the end sequents of γ_n. More formally we have

Proposition 6.0.14. *[6] Proposition* 6.5.3. *Let $(\varphi_n)_{n\in\mathbb{N}}$ be a sequence of proofs in Φ^s. Assume that there exists an elementary function f and a sequence of resolution refutations $(\gamma_n)_{n\in\mathbb{N}}$ of $(CL(\varphi_n))_{n\in\mathbb{N}}$ s.t.*

$$l(\gamma_n) \leq f(\|\varphi_n\|)$$

for $n \in \mathbb{N}$. Then there exists an elementary function g and a sequence of CERES normal forms φ_n^* of φ_n s.t.

$$\|\varphi_n^*\| \leq g(\|\varphi_n\|).$$

Proof. For the proof we refer to [6] Proposition 6.5.3. □

The following proposition shows that in the Statman proof sequence the lengths of resolution refutations of the characteristic clause sets are the main source of complexity in the CERES method.

Proposition 6.0.15. *[6] Proposition* 6.5.4. *Let* $(\gamma_n)_{n\in\mathbb{N}}$ *be the sequence of proofs of* $(S_n)_{n\in\mathbb{N}}$ *defined in [6] Section* 4.3 *and let* $(\rho_n)_{n\in\mathbb{N}}$ *be a sequence of resolution refutations of the sequence of clause sets* $(CL(\gamma_n))_{n\in\mathbb{N}}$*. Then* $(l(\rho_n))_{n\in\mathbb{N}}$ *is nonelementary in* $(PC^{\mathscr{A}_e}(S_n))_{n\in\mathbb{N}}$*.*

Proof. For the proof we refer to [6] Proposition 6.5.4. □

CHAPTER 7

Skolem-free CERES-Method and Herbrand Sequent Extraction

In this chapter we will state the main results of this thesis, which are on the one hand to gain a Skolem-free CERES method for first-order logic and on the other hand speeding-up the process of Herbrand sequent extraction. In the original CERES-method, Herbrand sequents are extracted out of the ACNF. This however we want to overcome, since in the Skolem-free CERES-method, it becomes complicated and expensive to compute the CERES-normal form out of the resolution refutation and the corresponding projections. This lead to a novel method, in the sense, that we extract the Herbrand sequents out of the resolution refutation and the corresponding projections. This makes the construction of the ACNF obsolete for us.

In Section 7.1 we will introduce a Skolem-free CERES method for first-order logic. In Section 7.2 we will show how Herbrand sequents can be extracted and how to extract Herbrand sequents out of the resolution refutation and the corresponding projections, instead of the CERES-normal form. We will also show that we can speed-up the process of Herbrand sequent extraction in Section 7.3. Finally in Section 7.4 we will give a complexity analysis of our new method.

7.1 Skolem-free CERES-method in first-order logic

Since CERES in first-order logic is restricted to Skolemized proofs, the aim of this thesis is to generalize the method such that the new method also works in the presence of strong quantifiers in the end-sequent, yielding a Skolem-free CERES method for first-order logic. In the original CERES-method for first-order logic, the derivation we get as result after cut-elimination can be transformed into another of the original un-Skolemized sequent. The replacement of the Skolem functions by the original quantifiers in the resulting CERES-normal form may lead to an exponential increase in terms of the symbolic complexity of the original end-sequent and of the CERES-normal form. Hence, the benefit of a Skolem-free CERES-method is that we do not need a de-Skolemization.

Our CERES-method is a mix of the original CERES-method and CERES$^\omega$. In particular, from the ordinary CERES-method we need:

- the resolution Calculus $\mathscr{R}$

and from CERES$^\omega$ we need:

- the calculus $\mathbf{LK}_{sk}$,
- the definition of $\mathbf{LK}_{skc}$-trees,
- cut-elimination for $\mathbf{LK}_{sk}$.

Hence, the CERES-method we use is very similar to CERES$^\omega$, except that we introduce a method for first-order logic and therefore skip in CERES$^\omega$ everything related to higher-order logic. We also do not need the resolution calculus $\mathscr{R}_{al}$ introduced for CERES$^\omega$ (since this is a resolution calculus for higher-order logic), but restrict our method to the resolution calculus for first-order logic, used in the original CERES-method.

All the definitions, propositions, lemmas and theorems that hold for the resolution calculus, as explained in Section 2.3, and for CERES$^\omega$, as explained in Chapter 5, also hold for our CERES-method.

We will not state the definitions of the resolution calculus, the calculus $\mathbf{LK}_{sk}$, of $\mathbf{LK}_{skc}$-trees and cut-elimination for $\mathbf{LK}_{sk}$ here again, instead we refer to the definitions in previous sections. We give a rough outline on the main steps of our method:

1. let π be a regular **LK**-proof of S containing cuts
2. transform π into a regular $\mathbf{LK}_{skc}$-proof φ of S, this can be done as explained in Lemma 5.4.2
3. construct the characteristic sequent set $CS_\rho(\varphi)$ and the projections $\mathscr{P}_\rho(\varphi)$, where ρ denotes the last inference in φ, as defined in Definition 5.4.2
4. construct an **LK**-refutation of the reduct of the characteristic sequent set $CS_\rho(\varphi)$, as explained in Proposition 5.4.3
5. transform the restricted $\mathbf{LK}_{skc}$-proof φ of S into an $\mathbf{LK}_{sk}$-proof φ' of S, as explained in Theorem 5.4.8
6. extract the Herbrand sequents from the $\mathbf{LK}_{sk}$-proof φ' of S, instead of generating the ACNF

Note, that we do not generate the ACNF in our CERES-method. The reason is that we do not need it for the extraction of Herbrand sequents. Also, in the Skolem-free CERES-method, it becomes complicated to compute the CERES-normal form out of the resolution refutation and the corresponding projections.

In the next section, we will show how Herbrand sequents are extracted. Furthermore, we will define a method on how to extract them out of the resolution refutation and the projections.

7.2 Extracting Herbrand Sequents

Cut-free proofs are not the only important result of the CERES method, because a long proof without cuts is not always useful. Instead we focus on the extraction of Herbrand sequents. Indeed, in many applications (in particular in the interpretation of mathematical proofs) you do not even need a cut-free proof, but the extraction of Herbrand sequents is of major importance.

The second main result of this thesis is the analysis of the Skolem-free CERES method and speeding-up the process of Herbrand sequent extraction. Since in the new method we will not generate a CERES-normal form for the extraction, but will extract the Herbrand sequents out of the resolution refutation and the corresponding projections, the Herbrand sequent extraction will result in a much faster method compared to the extraction in the original CERES method, because in the Skolem-free method the generation of the CERES-normal form is very complicated and expensive. The following definitions are taken from [6].

Definition 7.2.1. Instantiation sequent. Let $S : A_1, ..., A_n \vdash B_1, ..., B_m$ be a weakly quantified sequent. Let A_i^-, B_j^- be the formulas A_i, B_j after omission of the quantifier occurrences. For every i, j let $\vec{A_i}, \vec{B_j}$ be sequences of instances of A_i^- and B_j^-, respectively. Then any permutation of the sequent

$$S' : \vec{A_1}, ..., \vec{A_n} \vdash \vec{B_1}, ..., \vec{B_m}$$

is called an instantiation sequent of S.

Definition 7.2.2. $\mathscr{A}$-validity. Let $\mathscr{A}$ be an axiom set. A sequent S is called $\mathscr{A}$-valid if $\mathscr{A} \models S$.

Definition 7.2.3. Herbrand sequent. Let $\mathscr{A}$ be an axiom set and let S be an $\mathscr{A}$-valid sequent. An instantiation sequent S' of S is called an $\mathscr{A}$-Herbrand sequent of S if S' is $\mathscr{A}$-valid. If $\mathscr{A}$ is the standard axiom set then S' is called a Herbrand sequent of S.

The following example is taken from [6] Example 4.2.2.

Example 7.2.1. Let $S = P(a), (\forall x)(P(x) \to P(f(x))) \vdash (\exists y)P(f(f(y)))$. Then

$$S' : P(a), P(a) \to P(f(a)) \vdash P(f(f(x))), P(f(f(a)))$$

is an instantiation sequent of S, but not a Herbrand sequent of S.

$$S' : P(a), P(a) \to P(f(a)), P(f(a)) \to P(f(f(a))) \vdash P(f(f(a)))$$

is a Herbrand sequent of S.

The method for Herbrand sequent extraction used in the following is a constructive one, which obtains Herbrand sequents S' from **LK**-proofs φ of S. These Herbrand sequents can be directly obtained from proofs of arbitrary weakly quantified sequents. To simplify the construction of the Herbrand sequents, it is restricted here to prenex sequents only. The method is described in [6] Chapter 4.2.

The method in [6] describes the extraction of Herbrand sequents from the ACNF. However, it is possible to obtain them without constructing the ACNF. A Herbrand sequent of an ACNF can also be composed out of the Herbrand sequents of the projections, after deleting clause parts. This means that a resolution refutation of the clause set and the projections are sufficient for the extraction of Herbrand sequents and the construction of an ACNF is no longer needed.

The method in [6] analyses a proof φ of a sequent S and checks for each formula A occurring at a position μ in S if it is quantifier free. If it is, then $q(\varphi, \mu) = A$, otherwise $q(\varphi, \mu)$ is defined as a multi-set of all ancestors B of A in φ s.t. B is quantifier-free and is the auxiliary formula of a quantifier inference. If such an ancestor B does not exists, $q(\varphi, \mu)$ is defined as the empty sequence. Then the Herbrand sequent can be obtained by constructing $H(\varphi) = q(\varphi, \mu_1), ..., q(\varphi, \mu_n) \vdash q(\varphi, \nu_1), ..., q(\varphi, \nu_m)$ for an end-sequent $S = A_1, ..., A_n \vdash B_1, ..., B_m$ and μ_i being the occurrence of A_i and ν_i of B_i, respectively. Double occurrences of formulas in $H(\varphi)$ are omitted.

7.2.1 Extracting Herbrand sequents from projections

Given a proof φ of a prenex sequent S, the resolution refutation R of the characteristic clause set $CL(\varphi)$ and the projections $\varphi[C_1]$ ... $\varphi[C_n]$ for clauses C_1 ... $C_n \in CL(\varphi)$ the Herbrand sequents $H(\varphi[C_i])$, for $1 \leq i \leq n$, can be composed in the following way:
First, analyse the original proof φ of S and highlight all ancestors of the cut-formula. Then for each projection $\varphi[C_i]$ do the following:

- the highlighting of ancestors of the cut-formula remains in the projection, if necessary highlight new ancestors of the cut-formula (the set of all ancestors of cut formulas in φ is $\mathrm{C}(\varphi)$)
- remove all formulas in the end-sequent $S_{\varphi[C_i]}$ of $\varphi[C_i]$ that are contained in $\mathrm{C}(\varphi)$
- compute $H(\varphi[C_i]\sigma)$ out of the remaining formulas in $S_{\varphi[C_i]}$ in the usual way, where σ is the substitution used in the ground resolution refutation. In general there are many ground substitutions $\varphi[C_i]\sigma$ necessary. The number of these instances depends on the number of instantiations of C_i in the resolution proof R. In fact there is a unique σ but several copies of $\varphi[C_i]$.

Note that the only difference between this method and the original one is, that for the computation of the Herbrand sequent we only consider formulas in the end-sequents of the projections that are no ancestors of cut-formulas. To clarify the method, have a look on the examples below.

Now the final Herbrand sequent $H^*(\varphi)$ can be composed out of the Herbrand sequents of the projections.

$$H^*(\varphi) = H(\varphi[C_1]) \otimes H(\varphi[C_2]) \otimes \ \ldots \ \otimes H(\varphi[C_n])$$

for n projections. Finally, remove double occurrences of formulas in $H^*(\varphi)$. The composed $H^*(\varphi)$ is a Herbrand sequent of the ACNF of φ and equal to the Herbrand sequent constructed directly from the ACNF with the same method.

7.2.2 Examples

Example 1.
Consider the following proof φ of the sequent $A, \neg A \vee B \vdash B, C$ (the cut formula is orange):

$$\dfrac{\dfrac{\dfrac{\dfrac{A \vdash A}{\neg A, A \vdash}\neg_l \qquad B \vdash B}{A, \neg A \vee B \vdash B}\vee_l}{A, \neg A \vee B \vdash B \vee C}\vee_r \qquad \dfrac{B \vdash B \qquad C \vdash C}{B \vee C \vdash B, C}\vee_l}{A, \neg A \vee B \vdash B, C}\mathit{cut}$$

$CL(\varphi) = \{\vdash B\ ;\ B \vdash\ ;\ C \vdash\}$.
The resolution refutation is:

$$\dfrac{\vdash B \qquad B \vdash}{\vdash} R$$

Hence, we need to consider the projections of $\vdash B$ and $B \vdash$:
$\varphi[\vdash B]$:

$$\dfrac{\dfrac{\dfrac{\dfrac{A \vdash A}{\neg A, A \vdash}\neg_l \qquad B \vdash B}{A, \neg A \vee B \vdash B}\vee_l}{A, \neg A \vee B \vdash B, C}w_r}{A, \neg A \vee B \vdash B, C, B}w_r$$

$\varphi[B \vdash]$:

$$\dfrac{\dfrac{\dfrac{B \vdash B}{B \vdash B, C}w_r}{\neg A \vee B, B \vdash B, C}w_l}{A, \neg A \vee B, B \vdash B, C}w_l$$

Now consider the ACNF:

$$\dfrac{\dfrac{\begin{array}{c}\varphi[\vdash B]\\ A, \neg A \vee B \vdash B, C, B\end{array} \qquad \begin{array}{c}\varphi[B \vdash]\\ A, \neg A \vee B, B \vdash B, C\end{array}}{A, \neg A \vee B, A, \neg A \vee B \vdash B, C, B, C}\mathit{cut}}{A, \neg A \vee B \vdash B, C}c_l^*, c_r^*$$

Now compute the Herbrand sequents:

$$H(ACNF) = A, \neg A \vee B \vdash B, C$$

$$H(\varphi[\vdash B]) = A, \neg A \vee B \vdash C, B$$
$$H(\varphi[B \vdash]) = A, \neg A \vee B \vdash B, C$$

If you now compute

$$H^*(\varphi) = H(\varphi[\vdash B]) \times H(\varphi[B \vdash])$$

and remove double occurrences of formulas you get

$$H^*(\varphi) = A, \neg A \vee B, A, \neg A \vee B \vdash C, B, B, C = A, \neg A \vee B \vdash B, C$$

and this is equal to $H(ACNF)$.

Obviously, the Herbrand sequent is identical to the end-sequent of the proof.

Example 2.
Now consider a more complicated proof φ of the sequent $Pa, \forall x(Px \to Pfx) \vdash \exists z Pf^4 z$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P\alpha \vdash P\alpha \qquad \dfrac{Pf\alpha \vdash Pf\alpha \qquad Pf^2\alpha \vdash Pf^2\alpha}{Pf\alpha, Pf\alpha \to Pf^2\alpha \vdash Pf^2\alpha}\to_l}{P\alpha, P\alpha \to Pf\alpha, Pf\alpha \to Pf^2\alpha \vdash Pf^2\alpha}\to_l}{P\alpha, \forall x(Px \to Pfx), \forall x(Px \to Pfx) \vdash Pf^2\alpha}\forall_l, \forall_l}{\forall x(Px \to Pfx), \forall x(Px \to Pfx) \vdash P\alpha \to Pf^2\alpha}\to_r}{\forall x(Px \to Pfx) \vdash P\alpha \to Pf^2\alpha}c_l}{\forall x(Px \to Pfx) \vdash \forall x(Px \to Pf^2x)}\forall_r \qquad \dfrac{\dfrac{\dfrac{\dfrac{Pa \vdash Pa \qquad \dfrac{Pf^2a \vdash Pf^2a \qquad Pf^4a \vdash Pf^4a}{Pf^2a, Pf^2a \to Pf^4a \vdash Pf^4a}\to_l}{Pa, Pa \to Pf^2a, Pf^2a \to Pf^4a \vdash Pf^4a}\to_l}{Pa, \forall x(Px \to Pf^2x), \forall x(Px \to Pf^2x) \vdash Pf^4a}\forall_l, \forall_l}{Pa, \forall x(Px \to Pf^2x) \vdash Pf^4a}c_l}{Pa, \forall x(Px \to Pf^2x) \vdash \exists z Pf^4z}\exists_r}{Pa, \forall x(Px \to Pfx) \vdash \exists z Pf^4z}cut$$

$CL(\varphi) = \{P\alpha \vdash Pf^2\alpha;\ Pf^4a \vdash;\ \vdash Pa;\ Pf^2a \vdash Pf^2a\}$.
The resolution refutation is:

$$\dfrac{\dfrac{P\alpha \vdash Pf^2\alpha \qquad \vdash Pa}{\vdash Pf^2a}R\{\alpha \leftarrow a\} \qquad \dfrac{Pf^4a \vdash \qquad P\alpha \vdash Pf^2\alpha}{Pf^2a \vdash}R\{\alpha \leftarrow f^2a\}}{\vdash}R$$

Hence, we need to consider the projections of $P\alpha \vdash Pf^2\alpha$, $\vdash Pa$ and $Pf^4a \vdash$:
$\varphi[P\alpha \vdash Pf^2\alpha]$:

$$\dfrac{\dfrac{\dfrac{\dfrac{P\alpha \vdash P\alpha \qquad \dfrac{Pf\alpha \vdash Pf\alpha \qquad Pf^2\alpha \vdash Pf^2\alpha}{Pf\alpha, Pf\alpha \to Pf^2\alpha \vdash Pf^2\alpha}\to_l}{P\alpha, P\alpha \to Pf\alpha, Pf\alpha \to Pf^2\alpha \vdash Pf^2\alpha}\to_l}{P\alpha, \forall x(Px \to Pfx), \forall x(Px \to Pfx) \vdash Pf^2\alpha}\forall_l, \forall_l}{P\alpha, \forall x(Px \to Pfx) \vdash Pf^2\alpha}c_l}{P\alpha, Pa, \forall x(Px \to Pfx) \vdash Pf^2\alpha, \exists z Pf^4z}w_l, w_r$$

$\varphi[\vdash Pa]$:

$$\frac{Pa \vdash Pa}{Pa, \forall x(Px \to Pfx) \vdash Pa, \exists z Pf^4 z}\; w_l, w_r$$

$\varphi[Pf^4 a \vdash]$:

$$\frac{\dfrac{Pf^4 a \vdash Pf^4 a}{Pf^4 a \vdash \exists z Pf^4 z}\; \exists_r}{Pf^4 a, Pa, \forall x(Px \to Pfx) \vdash \exists z Pf^4 z}\; w_l$$

Now consider the ACNF:

$$\frac{\dfrac{\overset{\varphi_1}{Pa, \forall x(Px \to Pfx), \forall x(Px \to Pfx), \vdash \exists z Pf^4 z, \exists z Pf^4 z, Pf^2\alpha} \qquad \overset{\varphi_2}{Pa, Pa, Pf^2 a, \forall x(Px \to Pfx), \forall x(Px \to Pfx), \vdash \exists z Pf^4 z, \exists z Pf^4 z}}{Pa, Pa, Pa, \forall x(Px \to Pfx), \forall x(Px \to Pfx), \forall x(Px \to Pfx), \forall x(Px \to Pfx), \vdash \exists z Pf^4 z, \exists z Pf^4 z, \exists z Pf^4 z, \exists z Pf^4 z}\; cut}{Pa, \forall x(Px \to Pfx), \vdash \exists z Pf^4 z}\; c_l^*, c_r^*$$

φ_1 :

$$\frac{\overset{\varphi[P\alpha \vdash Pf^2\alpha](\alpha \leftarrow a)}{Pa, Pa, \forall x(Px \to Pfx), \vdash \exists z Pf^4 z, Pf^2\alpha} \qquad \overset{\varphi[\vdash Pa)}{Pa, \forall x(Px \to Pfx), \vdash \exists z Pf^4 z, Pa}}{Pa, \forall x(Px \to Pfx), \forall x(Px \to Pfx), \vdash \exists z Pf^4 z, \exists z Pf^4 z, Pf^2\alpha}\; cut$$

φ_2 :

$$\frac{\overset{\varphi[Pf^4 a \vdash]}{Pf^4 a, Pa, \forall x(Px \to Pfx), \vdash \exists z Pf^4 z} \qquad \overset{\varphi[P\alpha \vdash Pf^2\alpha](\alpha \leftarrow f^2 a)}{Pf^2 a, Pa, \forall x(Px \to Pfx), \vdash \exists z Pf^4 z, Pf^4 a}}{Pa, Pa, Pf^2 a, \forall x(Px \to Pfx), \forall x(Px \to Pfx), \vdash \exists z Pf^4 z, \exists z Pf^4 z}\; cut$$

Now compute the Herbrand sequents:

$$H(ACNF) = Pa, Pa \to Pfa, Pfa \to Pf^2 a, Pf^2 a \to Pf^3 a, Pf^3 a \to Pf^4 a \vdash Pf^4 a$$

$$H(\varphi[P\alpha \vdash Pf^2\alpha](\alpha \leftarrow a)) = Pa, Pa \to Pfa, Pfa \to Pf^2 a \vdash$$
$$H(\varphi[P\alpha \vdash Pf^2\alpha](\alpha \leftarrow f^2 a)) = Pa, Pf^2 a \to Pf^3 a, Pf^3 a \to Pf^4 a \vdash$$
$$H(\varphi[\vdash Pa]) = Pa \vdash$$
$$H(\varphi[Pf^4 a \vdash]) = Pa \vdash Pf^4 a$$

If you now compute

$$H^*(\varphi) = H(\varphi[P\alpha \vdash Pf^2\alpha](\alpha \leftarrow a)) \otimes H(\varphi[P\alpha \vdash Pf^2\alpha](\alpha \leftarrow f^2 a)) \otimes H(\varphi[\vdash Pa]) \otimes H(\varphi[Pf^4 a \vdash])$$

and remove double occurrences of formulas you get

$$H^*(\varphi) = Pa, Pa \to Pfa, Pfa \to Pf^2 a, Pf^2 a \to Pf^3 a, Pf^3 a \to Pf^4 a \vdash Pf^4 a$$

.

7.2.3 Herbrand sequents can be constructed from projections and resolution refutations

Theorem 7.2.1. *Herbrand sequent from projections and resolution refutations. Let φ be a proof of a prenex sequent S, $CL(\varphi)$ the characteristic clause set of φ, $H(ACNF)$ the Herbrand sequent extracted out of the ACNF of φ and $H^*(\varphi)$ the Herbrand sequent constructed out of the projections $\varphi[C_i]$ for $C_i \in CL(\varphi)$ for $1 \leq i \leq n$ and n the number of clauses. Then $H(ACNF)$ contains all formulas that are contained in $H^*(\varphi)$ and vice versa, i.e. $H^*(\varphi) = H(ACNF)$*

Proof. By induction on the number of nodes in the resolution tree.
Let $o(R)$ be the number of nodes in the resolution tree R.

Base case: $o(R) = 3$. Then the resolution refutation has the following structure

$$\frac{C_1 \qquad C_2}{\vdash} R(\sigma)$$

where C_1, C_2 and $\vdash$ are the three nodes and C_1 and C_2 have an appropriate structure (for example $C_1 = \ \vdash C$ and $C_2 = C \vdash$), where $C_1, C_2 \in CL(\varphi)$ and σ is a possible substitution. The $ACNF$ has the following structure

$$\cfrac{\cfrac{\begin{matrix}\varphi[C_1]\sigma \\ \cdots\end{matrix} \qquad \begin{matrix}\varphi[C_2]\sigma \\ \cdots\end{matrix}}{S'}\, cut}{S}\, c^* : l, r$$

Now it is easy to see that the Herbrand sequent computed out of the $ACNF$, $H(ACNF)$, and the one constructed out of the projections, $H^*(\varphi)$, are equal. For the construction of $H(ACNF)$ we only consider formulas in the end-sequent S. Let A be such a formula. Then there are three cases:

1. A is quantifier-free. Since all inferences between the projections and S are either cut-inferences or contractions, A is either contained in the end-sequent of $\varphi[C_1]\sigma$ or $\varphi[C_2]\sigma$ or is duplicated via $c : l$ or $c : r$ and contained in the end-sequents of both projections. Therefore A is considered in the construction of $H(ACNF)$ and of $H(\varphi[C_1]\sigma)$ or $H(\varphi[C_1]\sigma)$ or both of them. Hence A is considered in $H(\varphi[C_1]\sigma) \otimes H(\varphi[C_1]\sigma) = H^*(\varphi)$.

2. A is not quantifier-free and the sequence of ancestors B of A, s.t. B is quantifier-free and the auxiliary formula of some quantifier-inference, is empty. Then there is no such ancestor B of A in the $ACNF$, but then it trivially follows that there is no such ancestor B of A in $\varphi[C_1]\sigma$ and $\varphi[C_2]\sigma$, hence A is neither considered in the construction of $H(ACNF)$, nor in the construction of $H(\varphi[C_1]\sigma)$ and $H(\varphi[C_2]\sigma)$. Therefore A is also not considered in $H(\varphi[C_1]\sigma) \otimes H(\varphi[C_1]\sigma) = H^*(\varphi)$.

3. A is not quantifier-free and $H(ACNF)$ contains a sequence of ancestors B of A, s.t. B is quantifier-free and the auxiliary formula of some quantifier-inference. Each

such ancestor B has to be the auxiliary formula of a quantifier-inference in one of the two projections, since $\varphi[C_1]\sigma$ and $\varphi[C_2]\sigma$ are the only subproofs of the $ACNF$ that could contain quantifier-inferences. Therefore, all ancestors B of A that are contained in $H(ACNF)$ are also contained in $H(\varphi[C_1]\sigma)$ or $H(\varphi[C_2]\sigma)$ or both, and hence also contained in $H(\varphi[C_1]\sigma) \otimes H(\varphi[C_1]\sigma) = H^*(\varphi)$.

Now we only need to prove, that there are no additional formulas in the end-sequents of $\varphi[C_1]\sigma$ and $\varphi[C_2]\sigma$, that are considered in the construction of $H(\varphi[C_1]\sigma)$ and $H(\varphi[C_2]\sigma)$ and therefore are contained in $H^*(\varphi)$, that are not formulas of the end-sequent S. If this is shown, $H(\varphi[C_1]\sigma)$ and $H(\varphi[C_2]\sigma)$ consider the same formulas for their construction as $H(ACNF)$ and therefore $H^*(\varphi) = H(ACNF)$. But this is trivial, since all formulas in the end-sequents of $\varphi[C_1]\sigma$ and $\varphi[C_2]\sigma$ are either contained in the end-sequent S or ancestors of cut-formulas. So the only formulas in the end-sequents of the projections that are not contained in S are ancestors of cut-formulas, but these formulas are not used in the construction of $H(\varphi[C_1]\sigma)$ or $H(\varphi[C_2]\sigma)$, by the very definition of the method for the Herbrand sequent extraction. Hence, $H^*(\varphi) = H(ACNF)$.

(IH): Assume that for $o(R) \leq n$ it holds that $H^*(\varphi) = H(ACNF)$.

Note that we always have to add 2 nodes to the tree, in order to get a larger resolution tree, since adding just one node to the tree would not make any sense in a resolution refutation. Now consider $o(R) = n + 2$. We want to show that for $n + 2$ nodes in the resolution tree it also holds that $H^*(\varphi_{n+2}) = H(ACNF_{n+2})$.
The uppermost part of the resolution refutation has the following structure

$$\cfrac{\cfrac{C_{n+2} \qquad C_{n+1}}{C_n}\,R(\sigma) \qquad C_{n-1}}{\cfrac{\cdots}{\vdash}}\,R(\sigma')$$

where C_{n+1} and C_{n+2} are the new nodes and σ, σ_1, ... are possible substitutions (note that they can be different to each other). The $ACNF_{n+2}$ has the following structure

$$\cfrac{\cfrac{\cfrac{\varphi[C_{n+2}]\sigma}{\cdots} \qquad \cfrac{\varphi[C_{n+1}]\sigma}{\cdots}}{S'}\,cut \qquad \cfrac{\varphi[C_{n-1}]\sigma'}{\cdots}}{\cfrac{\cdots}{S}}\,cut$$

Note that the lower part of the resolution tree (below the uppermost R) contains n nodes and we know that for this part $H^*(\varphi) = H(ACNF)$ holds.

Therefore we have to show that we can consider C_n in the resolution tree as last node and show that the Herbrand sequent of S' is equal to $H(\varphi[C_{n+1}]\sigma) \otimes H(\varphi[C_{n+2}]\sigma)$. The last node of the resolution tree is then

$$\cfrac{C_{n+2} \qquad C_{n+1}}{C_n}\,R(\sigma)$$

and for the resulting resolution tree we have that $o(R) = n$. The resolution tree now has the following structure

$$\cfrac{C_n \qquad C_{n-1}}{\cfrac{\cdots}{\vdash}} \; R(\sigma')$$

The only difference in the $ACNF$ is, that we do not look at the projections of $\varphi[C_{n+2}]$ and $\varphi[C_{n+1}]$ but at the proof-tree of S' as an own node, containing $\varphi[C_{n+2}]$ and $\varphi[C_{n+1}]$ as subproofs with a cut-inference combining them.

$$\cfrac{S' \qquad \begin{matrix}\varphi[C_{n-1}]\sigma' \\ \cdots\end{matrix}}{\cfrac{\cdots}{S}} \; cut$$

Since for the modified resolution tree R we have $o(R) = n$, we can use the induction hypothesis and conclude that $H^*(\varphi) = H(ACNF)$. It remains to prove that $H(\varphi[C_{n+2}]\sigma) \times H(\varphi[C_{n+1}]\sigma)$ actually is the Herbrand sequent of S'. Again, for the construction of the Herbrand sequent of S', call it $H(S')$, we only consider formulas in the end-sequent S' that are not ancestors of cut-formulas. Then the proof is trivial, since this is just the base case. All formulas considered in the construction of $H(S')$ are also considered in the construction of $H(\varphi[C_{n+2}]\sigma) \times H(\varphi[C_{n+1}]\sigma)$. The proof proceeds analogously to the proof of the base case. Hence we conclude for the resolution tree with n nodes

$$H(ACNF) = H(S') \otimes H(\varphi[C_{n-1}]\sigma') \otimes \; \ldots \; = H^*(\varphi)$$

and for the resolution tree with $n+2$ nodes we replace $H(S')$ by $H(\varphi[C_{n+2}]) \otimes H(\varphi[C_{n+1}])$, resulting in

$$H(ACNF_{n+2}) = H(\varphi[C_{n+2}]\sigma) \otimes H(\varphi[C_{n+1}]\sigma) \otimes H(\varphi[C_{n-1}]\sigma') \otimes \; \ldots \; = H^*(\varphi_{n+2})$$

□

7.3 Herbrand sequent extraction with $\mathbf{LK}_{sk}$-proofs

In [22] and [13] it is shown that all $\mathbf{LK}_{sk}$-proofs can be translated into **LK**-proofs, which are needed for the construction of an ACNF. The proof is effective and based on permutations of inferences in $\mathbf{LK}_{sk}$-trees. Since we are not interested in the ACNF, but in Herbrand sequents, it would be of great interest to skip the transformation into a **LK**-proof and extract the Herbrand sequent out of the $\mathbf{LK}_{sk}$-proof. This would speed up the Herbrand sequents extraction by omitting superfluous proof-transformations.

In order to show that Herbrand sequents can be extracted out of an $\mathbf{LK}_{sk}$-proof, we first need to analyse the transformations described in [22] and [13]. To show how to permute unary and binary inferences the two rewrite relations $\rhd_u$ and $\rhd_b$ on $\mathbf{LK}_{sk}$-trees are introduced. In this section we will show that $\rhd_u$ and $\rhd_b$ do not influence the formulas that are used in the construction of the Herbrand sequent, i.e. that for the Herbrand sequent extraction the transformation into **LK**-proofs can be omitted. First we need a definition:

Definition 7.3.1. The relation $\rhd_{u/b}$. Let $\rhd_{u/b}$ be defined as the transitive and reflexive closure of the relations $\rhd_u^1 \cup \rhd_b^1$ relation. Therefore, $\rhd_{u/b}$ comprises both, the $\rhd_u$ and the $\rhd_b$ relation.

Now consider a $\mathbf{LK}_{sk}$-proof φ and a $\mathbf{LK}$-proof φ' of an end-sequent S s.t. φ' is constructed out of φ by permutations of inferences. Therefore $\varphi \rhd_{u/b} \varphi'$ holds. Note that we only have to consider permutations that involve quantifier inferences. The reason is simple, for the extraction of the Herbrand sequent, we have to consider each formula A in the end-sequent S of φ' and analyse its structure. Three cases can arise:

1. A is quantifier-free and we do not analyse the proof any further. Since for each rewrite step $\varphi \rhd_u^1 \varphi'$ and $\varphi \rhd_b^1 \varphi'$ it holds that the end-sequent does not change (i.e. the end-sequent of φ is the same as the end-sequent of φ'), we conclude that if A is a quantifier-free formula in the end-sequent of φ', it is also a quantifier-free formula in the end-sequent of φ. Therefore we can extract A out of the end-sequent of φ and do not need a transformation into φ'.

2. A is not quantifier-free and there is no ancestor B of A s.t. B is quantifier-free and the auxiliary formula of a quantifier inference. Then there is no such ancestor B of A in φ neither, since inferences are only permuted but there are no new inferences introduced. So it is impossible to find a quantifier inference in φ with auxiliary formula B that is quantifier-free and an ancestor of A. We conclude that if there are no ancestors B of A in φ' that are quantifier-free and the auxiliary formula of a quantifier inference, then there are also no such ancestors in φ and again we can skip the transformation into φ' and only consider φ.

3. A is not quantifier-free and there is some ancestor B of A s.t. B is quantifier-free and the auxiliary formula of a quantifier inference. In this case we have quantifier-inferences in the proof-tree operating on ancestors of quantified end-sequent formulas and therefore we have to analyse permutations that involve these inferences, since they could affect formulas that are used in the construction of the Herbrand sequent.

Hence, we state the following:

Theorem 7.3.1. *If for some **LK**-proof π and $\mathbf{LK}_{sk}$-proof φ it holds that $\varphi \rhd_{u/b} \pi$ in $o(\varphi)$ rewrite steps, then the Herbrand sequent of π can already be extracted out of φ.*

Proof. By induction on the number of rewrite steps $o(\varphi)$ in $\rhd_{u/b}$.

Base case: $o(\varphi) = 1$ and either $\varphi \rhd_u^1 \pi$ or $\varphi \rhd_b^1 \pi$ holds.
Here we consider every possible rewrite step and show that it can be omitted.

For $\rhd_u$ we have to consider:
This relation is used to permute down a unary logical inference ρ over an inference σ, assuming that ρ and σ are independent. Assume ρ is a quantifier inference. This means

that it could be the case, that its auxiliary formula is a formula B which is an ancestor of an end-sequent formula A, s.t. B is quantifier-free. Let φ' be the subproof of φ that contains the inferences that are permuted, and π' the respective subproof of π. If $\varphi' \rhd_u^1 \pi'$ and ρ is a quantifier inference we have five cases:

1. σ is an unary logical inference and φ' is

$$\dfrac{\dfrac{F^{i_1}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, F^{\overline{i_1}}}{M^{i_3}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, M^{\overline{i_3}}}\,\rho}{M^{i_3}, N^{i_4}, \Gamma \vdash \Delta, N^{\overline{i_4}}, M^{\overline{i_3}}}\,\sigma$$

and π' is

$$\dfrac{\dfrac{F^{i_1}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, F^{\overline{i_1}}}{F^{i_1}, N^{i_4}, \Gamma \vdash \Delta, N^{\overline{i_4}}, F^{\overline{i_1}}}\,\sigma}{M^{i_3}, N^{i_4}, \Gamma \vdash \Delta, N^{\overline{i_4}}, M^{\overline{i_3}}}\,\rho$$

If σ is no quantifier inference, we just need to look at ρ. Therefore, M^{i_3} and $M^{\overline{i_3}}$ are quantified formulas and we assume that they are either end-sequent formulas or ancestors of quantified end-sequent formulas. We know that F^{i_1} and $F^{\overline{i_1}}$ are the ancestors of M^{i_3} and $M^{\overline{i_3}}$, that are auxiliary formulas of a quantifier inference. If F^{i_1} and $F^{\overline{i_1}}$ are quantifier-free, i.e. M^{i_3} or $M^{\overline{i_3}}$ contain only one quantifier, then they are used in the construction of the Herbrand sequent of π, otherwise not.

But we see that the same formulas are also used in φ in the construction of the Herbrand sequent. Here we also assume M^{i_3} and $M^{\overline{i_3}}$ to be quantified formulas either in the end-sequent of φ (which is the same end-sequent as in π) or ancestors of quantified end-sequent formulas. In φ' first the unary inference σ is applied, but this has no influence on M^{i_3} and $M^{\overline{i_3}}$. With the inference ρ we get F^{i_1} and $F^{\overline{i_1}}$, which are the auxiliary formulas of a quantifier inference. If they are quantifier-free, and since they are ancestors of quantified formulas in the end-sequent, they are used for the construction of the Herbrand sequent, just like in the Herbrand sequent of π. We also note here that no new quantifier inference is introduced, such that the Herbrand sequent could change.

Now assume σ is a quantifier inference, too. In this case we also have to consider the auxiliary and main formulas of σ. We proceed in a similar way. In π' we assume N^{i_4} and $N^{\overline{i_4}}$ to be quantified formulas either in the end-sequent of π or ancestors of quantified end-sequent formulas. ρ has no influence on N^{i_4} and $N^{\overline{i_4}}$ and σ is a quantifier-inference with main formula N^{i_4} or $N^{\overline{i_4}}$ s.t. G^{i_2} or $G^{\overline{i_2}}$ is the auxiliary formula. If G^{i_2} or $G^{\overline{i_2}}$, respectively is quantifier-free, it is used in the construction of the Herbrand sequent.

But here again we see that the same formulas G^{i_2} and $G^{\overline{i_2}}$ are used in the construction of the Herbrand sequent of φ. N^{i_4} and $N^{\overline{i_4}}$ are quantified formulas and either contained in the end-sequent of φ or ancestors of quantified end-sequent formulas. σ is a quantifier-inference with auxiliary formula G^{i_2} or $G^{\overline{i_2}}$ which are ancestors of N^{i_4} and $N^{\overline{i_4}}$, hence if G^{i_2} or $G^{\overline{i_2}}$ are quantifier-free, they are used in the construction of the Herbrand sequent.

2. σ is a weakening inference and φ' is

$$\cfrac{\cfrac{F^{i_1}, \Gamma \vdash \Delta, F^{\overline{i_1}}}{M^{i_2}, \Gamma \vdash \Delta, M^{\overline{i_2}}}\ \rho}{N^{i_3}, M^{i_2}, \Gamma \vdash \Delta, M^{\overline{i_2}}, N^{\overline{i_3}}}\ \sigma$$

and π' is

$$\cfrac{\cfrac{F^{i_1}, \Gamma \vdash \Delta, F^{\overline{i_1}}}{N^{i_3}, F^{i_1}, \Gamma \vdash \Delta, F^{\overline{i_1}}, N^{i_3}}\ \sigma}{N^{i_3}, M^{i_2}, \Gamma \vdash \Delta, M^{\overline{i_2}}, N^{\overline{i_3}}}\ \rho$$

Assume M^{i_2} and $M^{\overline{i_2}}$ are quantified formulas and either they are formulas of the end-sequent of π or ancestors of quantified end-sequent formulas. ρ is a quantifier inference with auxiliary formula F^{i_1} or $F^{\overline{i_1}}$ and they are ancestors of M^{i_2} and $M^{\overline{i_2}}$. If the auxiliary formula F^{i_1} or $F^{\overline{i_1}}$ is quantifier-free, it is used in the construction of the Herbrand sequent.

Now consider φ'. Here again M^{i_2} and $M^{\overline{i_2}}$ are quantifier formulas and either they are formulas of the end-sequent of φ or ancestors of quantified end-sequent formulas. σ has no influence on M^{i_2} and $M^{\overline{i_2}}$ and ρ is a quantifier inference with auxiliary formula F^{i_1} or $F^{\overline{i_1}}$. If the auxiliary formula F^{i_1} or $F^{\overline{i_1}}$ is quantifier-free and since it is ancestor of either M^{i_2} or $M^{\overline{i_2}}$ and therefore ancestor of an end-sequent formula, we have that it is used in the construction of the Herbrand sequent of φ.

3. σ is a contraction inference and φ' is

$$\cfrac{\cfrac{F^{i_1}, G^{i_2}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, G^{\overline{i_2}}, F^{\overline{i_1}}}{M^{i_3}, G^{i_2}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, G^{\overline{i_2}}, M^{\overline{i_3}}}\ \rho}{M^{i_3}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, M^{\overline{i_3}}}\ \sigma$$

and π' is

$$\cfrac{\cfrac{F^{i_1}, G^{i_2}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, G^{\overline{i_2}}, F^{\overline{i_1}}}{F^{i_1}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, F^{\overline{i_1}}}\ \sigma}{M^{i_3}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, M^{\overline{i_3}}}\ \rho$$

Assuming that M^{i_3} and $M^{\overline{i_3}}$ are quantified formulas and either are in the end-sequent of π or ancestors of quantified end-sequent formulas, and ρ is a quantifier-inference with auxiliary formula F^{i_1} or $F^{\overline{i_1}}$, we have that F^{i_1} and $F^{\overline{i_1}}$ are ancestors of M^{i_3} and $M^{\overline{i_3}}$. If the auxiliary formula F^{i_1} or $F^{\overline{i_1}}$ is quantifier-free, it is used in the construction of the Herbrand sequent.

But the same formula is used in the construction of the Herbrand sequent of φ. Here we also assume that M^{i_3} and $M^{\overline{i_3}}$ are quantified formulas and either in the end-sequent of φ or ancestors of quantified end-sequent formulas. σ has no influence on M^{i_3} and $M^{\overline{i_3}}$ and ρ is a quantifier inference with auxiliary formula F^{i_1} or $F^{\overline{i_1}}$, which are ancestors of M^{i_3} and $M^{\overline{i_3}}$. If the auxiliary formula F^{i_1} or $F^{\overline{i_1}}$ is quantifier-free, it is used in the construction of the Herbrand sequent.

4. σ is a $\vee : l$ inference and φ' is

$$\dfrac{\dfrac{F^{i_1}, G_1, \Gamma \vdash \Delta, F^{\overline{i_1}}}{M^{i_2}, G_1, \Gamma \vdash \Delta, M^{\overline{i_2}}}\,\rho \qquad G_2, \Pi \vdash \Lambda}{G_1 \vee G_2, M^{i_2}, \Gamma, \Pi \vdash \Delta, \Lambda, M^{\overline{i_2}}}\,\sigma$$

and π' is

$$\dfrac{\dfrac{F^{i_1}, G_1, \Gamma \vdash \Delta, F^{\overline{i_1}} \qquad G_2, \Pi \vdash \Lambda}{G_1 \vee G_2, F^{i_1}, \Gamma, \Pi \vdash \Delta, \Lambda, F^{\overline{i_1}}}\,\sigma}{G_1 \vee G_2, M^{i_2}, \Gamma, \Pi \vdash \Delta, \Lambda, M^{\overline{i_2}}}\,\rho$$

If we assume M^{i_2} and $M^{\overline{i_2}}$ to be quantified formulas and contained in the end-sequent of π or ancestors of quantified end-sequent formulas, then F^{i_1} and $F^{\overline{i_1}}$ are their ancestors and one of them is the auxiliary formula of a quantifier inference. If the auxiliary formula F^{i_1} or $F^{\overline{i_1}}$ is quantifier-free, it is used in the construction of the Herbrand sequent of π.

If we now consider φ', it is easy to see that the same formula F^{i_1} or $F^{\overline{i_1}}$ is used in the construction of the Herbrand sequent of φ, if it is quantifier-free. Here we also assume M^{i_2} and $M^{\overline{i_2}}$ to be quantified formulas and either be end-sequent formulas or ancestors of quantifier end-sequent formulas. The $\vee : l$ inference does not change M^{i_2} and $M^{\overline{i_2}}$ but one of them is the main formula of ρ. F^{i_1} or $F^{\overline{i_1}}$ is the auxiliary formula and therefore ancestor of a quantifier formula in the end-sequent. Hence, if the auxiliary formula F^{i_1} or $F^{\overline{i_1}}$ is quantifier-free, it is used in the construction of the Herbrand sequent.

5. σ is again a $\vee : l$ inference and φ' is

$$\dfrac{G_1, \Gamma \vdash \Delta \qquad \dfrac{F^{i_1}, G_2, \Pi \vdash \Lambda, F^{\overline{i_1}}}{M^{i_2}, G_2, \Pi \vdash \Lambda, M^{\overline{i_2}}}\,\rho}{G_1 \vee G_2, M^{i_2}, \Gamma, \Pi \vdash \Delta, \Lambda, M^{\overline{i_2}}}\,\sigma$$

and π' is

$$\cfrac{\cfrac{G_1, \Gamma \vdash \Delta \qquad F^{i_1}, G_2, \Pi \vdash \Lambda, F^{\overline{i_1}}}{G_1 \vee G_2, F^{i_1}, \Gamma, \Pi \vdash \Delta, \Lambda, F^{\overline{i_1}}}\ \sigma}{G_1 \vee G_2, M^{i_2}, \Gamma, \Pi \vdash \Delta, \Lambda, M^{\overline{i_2}}}\ \rho$$

In π' F^{i_1} or $F^{\overline{i_1}}$ is the auxiliary formula of a quantifier inference. If M^{i_2} and $M^{\overline{i_2}}$ are quantified formulas in the end-sequent of π or quantified ancestors of quantified end-sequent formulas and F^{i_1} or $F^{\overline{i_1}}$ is quantifier-free, then the auxiliary formula F^{i_1} or $F^{\overline{i_1}}$ is used in the construction of the Herbrand sequent.

But in φ the same formula F^{i_1} or $F^{\overline{i_1}}$ is used for the construction of the Herbrand sequent. Again, M^{i_2} and $M^{\overline{i_2}}$ are quantified formulas in the end-sequent of φ or quantified ancestors of quantified end-sequent formulas and since $\vee : l$ does not chance M^{i_2} and $M^{\overline{i_2}}$ we only need to look at the quantifier inference. Its auxiliary formula is either F^{i_1} or $F^{\overline{i_1}}$ and if it is quantifier-free, it is used in the construction of the Herbrand sequent.

For $\rhd_b$ we consider the following:
This relation is used to permute $\vee : l$ inferences downwards. In $\mathbf{LK}_{sk}$ there are also binary inferences $\rightarrow : l$ and $\wedge : r$ but the analysis is similar, so we skip the proof for them. Also, we could get rid of $\rightarrow : l$, of course. In the following ρ is a $\vee : l$ inference and we only consider those cases, where σ is a quantifier inference.

1. φ' is

$$\cfrac{\cfrac{\cfrac{F_1, \Pi, \Gamma_1, G^{i_1} \vdash \Delta_1, G^{\overline{i_1}}, \Lambda \qquad F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda}{F_1 \vee F_2, \Pi, \Pi, \Gamma_1, \Gamma_2, G^{i_1} \vdash \Delta_1, G^{\overline{i_1}}, \Delta_2, \Lambda, \Lambda}\ \rho}{G^{i_1}, F_1 \vee F_2, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, G^{\overline{i_1}}}\ contr : *}{M^{i_2}, F_1 \vee F_2, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, M^{\overline{i_2}}}\ \sigma$$

and π' is

$$\cfrac{\cfrac{\cfrac{G^{i_1}, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda, G^{\overline{i_1}}}{M^{i_2}, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda, M^{\overline{i_2}}}\ \sigma \qquad F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda}{F_1 \vee F_2, M^{i_2}, \Pi, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda, M^{\overline{i_2}}}\ \rho}{F_1 \vee F_2, M^{i_2}, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, M^{\overline{i_2}}}\ contr : *$$

Assume M^{i_2} and $M^{\overline{i_2}}$ to be quantified formulas either in the end-sequent of π or ancestors of quantified formulas in the end-sequent. Since σ is a quantifier-inference with auxiliary formula G^{i_1} or $G^{\overline{i_1}}$ that are ancestors of M^{i_2} and $M^{\overline{i_2}}$ we have that if the auxiliary formula G^{i_1} or $G^{\overline{i_1}}$ is quantifier-free, then it is considered in the construction of the Herbrand sequent.

In φ the same formula is considered in the construction of the Herbrand sequent. Here, M^{i_2} and $M^{\overline{i_2}}$ again are assumed to be quantified formulas either in the end-sequent of φ or ancestors of quantified formulas in the end-sequent. On of them is the main formula of σ, which is a quantifier inference with auxiliary formula G^{i_1} or $G^{\overline{i_1}}$. So if the auxiliary formula G^{i_1} or $G^{\overline{i_1}}$ is quantifier-free, then it is used in the construction of the Herbrand sequent.

2. φ' is

$$\dfrac{\dfrac{\dfrac{F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda \qquad F_2, \Pi, \Gamma_2, G^{i_1} \vdash \Delta_2, \Lambda, G^{\overline{i_1}}}{F_1 \lor F_2, \Pi, \Pi, \Gamma_1, \Gamma_2, G^{i_1} \vdash \Delta_1, \Delta_2, \Lambda, \Lambda, G^{\overline{i_1}}}\,\rho}{G^{i_1}, F_1 \lor F_2, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, G^{\overline{i_1}}}\,contr: *}{M^{i_2}, F_1 \lor F_2, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, M^{\overline{i_2}}}\,\sigma$$

and π' is

$$\dfrac{\dfrac{F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda \qquad \dfrac{G^{i_1}, F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda, G^{\overline{i_1}}}{M^{i_2}, F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda, M^{\overline{i_2}}}\,\sigma}{F_1 \lor F_2, M^{i_2}, \Pi, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda, M^{\overline{i_2}}}\,\rho}{F_1 \lor F_2, M^{i_2}, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, M^{\overline{i_2}}}\,contr: *$$

Assume in π' M^{i_2} and $M^{\overline{i_2}}$ are quantified formulas that are either in the end-sequent of π or ancestors of quantified end-sequent formulas. We see that σ is a quantifier inference with main formula M^{i_2} or $M^{\overline{i_2}}$. The auxiliary formula of σ is then either G^{i_1} or $G^{\overline{i_1}}$. If the auxiliary formula G^{i_1} or $G^{\overline{i_1}}$ is quantifier-free, then ot is considered in the construction of the Herbrand sequent.

In φ we have the same formula considered in the construction of the Herbrand sequent. Here again M^{i_2} and $M^{\overline{i_2}}$ are assumed to be quantified formulas that are either in the end-sequent of φ or ancestors of quantified end-sequent formulas. σ has auxiliary formula G^{i_1} or $G^{\overline{i_1}}$ and if it is quantifier-free, it is an auxiliary formula of a quantifier-inference and ancestor of a quantified formula in the end-sequent, and therefore it is used in the construction of the Herbrand sequent.

3. φ' is

$$\dfrac{\dfrac{\dfrac{F_1, \Pi, G^{i_1}, \Gamma_1 \vdash \Delta_1, \Lambda, G^{\overline{i_1}} \qquad F_2, \Pi, G^{i_1}, \Gamma_2 \vdash \Delta_2, \Lambda, G^{\overline{i_1}}}{F_1 \lor F_2, \Pi, G^{i_1}, \Pi, G^{i_1}, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, G^{\overline{i_1}}, \Lambda, G^{\overline{i_1}}}\,\rho}{G^{i_1}, F_1 \lor F_2, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, G^{\overline{i_1}}}\,contr: *}{M^{i_2}, F_1 \lor F_2, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, M^{\overline{i_2}}}\,\sigma$$

and π' is

$$\dfrac{\dfrac{\dfrac{G^{i_1}, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda, G^{\overline{i_1}}}{M^{i_2}, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda, M^{\overline{i_2}}}\sigma \quad \dfrac{G^{i_1}, F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda, G^{\overline{i_1}}}{M^{i_2}, F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda, M^{\overline{i_2}}}\sigma}{F_1 \vee F_2, \Pi, M^{i_2}, \Pi, M^{i_2}, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, M^{\overline{i_2}}, \Lambda, M^{\overline{i_2}}}\rho}{F_1 \vee F_2, \Pi, M^{i_2}, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, M^{\overline{i_2}}}\mathit{contr} : *$$

Consider π' and assume M^{i_2} and $M^{\overline{i_2}}$ to be quantified formulas that either are end-sequent formulas of π or ancestors of quantified end-sequent formulas. If we now search for ancestors of M^{i_2} and $M^{\overline{i_2}}$ that are auxiliary formulas of quantifier inferences and quantifier-free, then we see that we find G^{i_1} and G^{i_1} or $G^{\overline{i_1}}$ and $G^{\overline{i_1}}$ as auxiliary formulas of quantifier inferences. If the auxiliary formulas are quantifier-free, then they are considered in the construction of the Herbrand sequent.

In φ' we also assume M^{i_2} and $M^{\overline{i_2}}$ to be quantified formulas that either are end-sequent formulas or ancestors of quantified end-sequent formulas. If their ancestors G^{i_1} and $G^{\overline{i_1}}$ are quantifier-free and since one of them is the auxiliary formula of a quantifier inference, we have that we consider the auxiliary formula G^{i_1} or $G^{\overline{i_1}}$ in the construction of the Herbrand sequent. Since in the Herbrand sequent we omit double occurrences of formulas, we have that the two extracted Herbrand sequents are equal.

Induction hypothesis: Assume that for $o(\varphi) = n$ rewrite steps it holds that for $\varphi \rhd^n_{u/b} \pi$ the Herbrand sequent of π can already be extracted out of φ and the transformation can be omitted.

Induction step: Show that (IH) holds for $o(\varphi) = n + 1$ rewrite steps.
Now the proof is simple. Consider the $n+1$ rewrite steps in $\varphi \rhd^{n+1}_{u/b} \pi$. The last rewrite step is a permutation of inferences in a subproof φ' of φ leading to a supbroof π' of π. For this last rewrite step we have $\varphi' \rhd^1_{u/b} \pi'$ and there are again the same cases as for the base case, which need to be considered. We have shown that every single rewrite step $\varphi' \rhd^1_{u/b} \pi'$ can be omitted. Hence we conclude that the last rewrite step $\varphi' \rhd^1_{u/b} \pi'$ can be omitted.

Therefore we have n rewrite steps left. But for $\varphi \rhd^n_{u/b} \pi$ we know by (IH) that the assumption holds. □

7.4 Complexity of proof-transformations

The complexity of cut-elimination is nonelementary in the size of the input proof and this holds for any cut-elimination method, therefore it holds for our CERES-method too. Since we do not generate the ACNF but extract the Herbrand sequents out of the projections and the resolution refutation and skip the transformation of a $\mathbf{LK}_{sk}$-proof into a $\mathbf{LK}$-proof, we are interested in the speed-up gained by omitting unnecessary transformations.

In this section we will analyse the complexity of the transformations from a $\mathbf{LK}_{sk}$-proof into a $\mathbf{LK}$-proof. Therefore we will analyse how the transformations $\rhd_u$ and $\rhd_b$ affect the proof-length of the proof. First we need some definitions. We will give two definitions for the relations $\rhd_u$ and $\rhd_b$, which are based on [13] Definition 21 and Definition 22.

Definition 7.4.1. The relation $\rhd_u$:
We permute a unary logical inference ρ over an inference σ, where ρ and σ are independent. In case 1, σ is an unary logical inference, in case 2 σ is a weakening inference, in case 3 σ is a contraction inference and in cases $4-5$ σ is an $\vee : l$ inference. Consider the relation $\rhd_u^1$ between $\mathbf{LK}_{sk}$-trees π and π':

1. π is

$$\dfrac{\dfrac{\begin{array}{c}(\varphi)\\ F^{i_1}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, F^{\overline{i_1}}\end{array}}{M^{i_3}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, M^{\overline{i_3}}}\,\rho}{M^{i_3}, N^{i_4}, \Gamma \vdash \Delta, N^{\overline{i_4}}, M^{\overline{i_3}}}\,\sigma$$

   and π' is

$$\dfrac{\dfrac{\begin{array}{c}(\varphi)\\ F^{i_1}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, F^{\overline{i_1}}\end{array}}{F^{i_1}, N^{i_4}, \Gamma \vdash \Delta, N^{\overline{i_4}}, F^{\overline{i_1}}}\,\sigma}{M^{i_3}, N^{i_4}, \Gamma \vdash \Delta, N^{\overline{i_4}}, M^{\overline{i_3}}}\,\rho$$

   then $\pi \rhd_u^1 \pi'$.

2. π is

$$\dfrac{\dfrac{\begin{array}{c}(\varphi)\\ F^{i_1}, \Gamma \vdash \Delta, F^{\overline{i_1}}\end{array}}{M^{i_2}, \Gamma \vdash \Delta, M^{\overline{i_2}}}\,\rho}{N^{i_3}, M^{i_2}, \Gamma \vdash \Delta, M^{\overline{i_2}}, N^{\overline{i_3}}}\,\sigma$$

   and π' is

$$\dfrac{\dfrac{\begin{matrix}(\varphi)\\ F^{i_1}, \Gamma \vdash \Delta, F^{\overline{i_1}}\end{matrix}}{N^{i_3}, F^{i_1}, \Gamma \vdash \Delta, F^{\overline{i_1}}, N^{i_3}}\,\sigma}{N^{i_3}, M^{i_2}, \Gamma \vdash \Delta, M^{\overline{i_2}}, N^{\overline{i_3}}}\,\rho$$

then $\pi \rhd_u^1 \pi'$.

3. π is

$$\dfrac{\dfrac{\begin{matrix}(\varphi)\\ F^{i_1}, G^{i_2}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, G^{\overline{i_2}}, F^{\overline{i_1}}\end{matrix}}{M^{i_3}, G^{i_2}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, G^{\overline{i_2}}, M^{\overline{i_3}}}\,\rho}{M^{i_3}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, M^{\overline{i_3}}}\,\sigma$$

and π' is

$$\dfrac{\dfrac{\begin{matrix}(\varphi)\\ F^{i_1}, G^{i_2}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, G^{\overline{i_2}}, F^{\overline{i_1}}\end{matrix}}{F^{i_1}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, F^{\overline{i_1}}}\,\sigma}{M^{i_3}, G^{i_2}, \Gamma \vdash \Delta, G^{\overline{i_2}}, M^{\overline{i_3}}}\,\rho$$

then $\pi \rhd_u^1 \pi'$.

4. π is

$$\dfrac{\dfrac{\begin{matrix}(\varphi_1)\\ F^{i_1}, G_1, \Gamma \vdash \Delta, F^{\overline{i_1}}\end{matrix}}{M^{i_2}, G_1, \Gamma \vdash \Delta, M^{\overline{i_2}}}\,\rho \qquad \begin{matrix}(\varphi_2)\\ G_2, \Pi \vdash \Lambda\end{matrix}}{G_1 \vee G_2, M^{i_2}, \Gamma, \Pi \vdash \Delta, \Lambda, M^{\overline{i_2}}}\,\sigma$$

and π' is

$$\dfrac{\dfrac{\begin{matrix}(\varphi_1)\\ F^{i_1}, G_1, \Gamma \vdash \Delta, F^{\overline{i_1}}\end{matrix} \qquad \begin{matrix}(\varphi_2)\\ G_2, \Pi \vdash \Lambda\end{matrix}}{G_1 \vee G_2, F^{i_1}, \Gamma, \Pi \vdash \Delta, \Lambda, F^{\overline{i_1}}}\,\sigma}{G_1 \vee G_2, M^{i_2}, \Gamma, \Pi \vdash \Delta, \Lambda, M^{\overline{i_2}}}\,\rho$$

then $\pi \rhd_u^1 \pi'$.

5. π is

$$\dfrac{\dfrac{(\varphi_1)}{G_1, \Gamma \vdash \Delta} \quad \dfrac{\dfrac{(\varphi_2)}{F^{i_1}, G_2, \Pi \vdash \Lambda, F^{\overline{i_1}}}}{M^{i_2}, G_2, \Pi \vdash \Lambda, M^{\overline{i_2}}}\,\rho}{G_1 \vee G_2, M^{i_2}, \Gamma, \Pi \vdash \Delta, \Lambda, M^{\overline{i_2}}}\,\sigma$$

and π' is

$$\dfrac{\dfrac{\dfrac{(\varphi_1)}{G_1, \Gamma \vdash \Delta} \quad \dfrac{(\varphi_2)}{F^{i_1}, G_2, \Pi \vdash \Lambda, F^{\overline{i_1}}}}{G_1 \vee G_2, F^{i_1}, \Gamma, \Pi \vdash \Delta, \Lambda, F^{\overline{i_1}}}\,\sigma}{G_1 \vee G_2, M^{i_2}, \Gamma, \Pi \vdash \Delta, \Lambda, M^{\overline{i_2}}}\,\rho$$

then $\pi \rhd_u^1 \pi'$.

Note that in case 3 in the definition above σ is a contraction left and right inference. Since such an inference is not defined in our calculus, we assume σ to be two inferences σ_1 and σ_2 where σ_1 is a c_l inference an σ_2 a c_r inference. The consecutive application of σ_1 and σ_2 yields σ.

The $\rhd_u$ relation is defined as the transitive and reflexive closure of the compatible closure of the $\rhd_u^1$ relation.

Definition 7.4.2. The relation $\rhd_b$:
Here we permute down a $\vee : l$ inference ρ. In cases $1-3$ σ is a unary logical inference, in case 4 σ is a weakening, in cases $5-6$ σ is a contraction inference and in cases $7-9$ σ is a $\vee : l$ inference.

1. π is

$$\dfrac{\dfrac{\dfrac{\dfrac{(\varphi_1)}{F_1, \Pi, \Gamma_1, G^{i_1} \vdash \Delta_1, G^{\overline{i_1}}, \Lambda} \quad \dfrac{(\varphi_2)}{F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda}}{F_1 \vee F_2, \Pi, \Pi, \Gamma_1, \Gamma_2, G^{i_1} \vdash \Delta_1, G^{\overline{i_1}}, \Delta_2, \Lambda, \Lambda}\,\rho}{G^{i_1}, F_1 \vee F_2, \Pi, \Pi', \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda', G^{\overline{i_1}}}\,c_*}{M^{i_2}, F_1 \vee F_2, \Pi, \Pi', \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda', M^{\overline{i_2}}}\,\sigma$$

and π' is

$$\dfrac{\dfrac{\dfrac{\dfrac{(\varphi_1)}{G^{i_1}, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda, G^{\overline{i_1}}}}{M^{i_2}, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda, M^{\overline{i_2}}}\,\sigma \quad \dfrac{(\varphi_2)}{F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda}}{F_1 \vee F_2, M^{i_2}, \Pi, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda, M^{\overline{i_2}}}\,\rho}{F_1 \vee F_2, M^{i_2}, \Pi, \Pi', \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda', M^{\overline{i_2}}}\,c_*$$

then $\pi \rhd_b^1 \pi'$.

2. π is

$$\dfrac{\dfrac{\dfrac{\begin{matrix}(\varphi_1) \\ F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda\end{matrix} \qquad \begin{matrix}(\varphi_2) \\ F_2, \Pi, \Gamma_2, G^{i_1} \vdash \Delta_2, \Lambda, G^{\overline{i_1}}\end{matrix}}{F_1 \lor F_2, \Pi, \Pi, \Gamma_1, \Gamma_2, G^{i_1} \vdash \Delta_1, \Delta_2, \Lambda, \Lambda, G^{\overline{i_1}}}\,\rho}{G^{i_1}, F_1 \lor F_2, \Pi, \Pi', \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda', G^{\overline{i_1}}}\,c_*}{M^{i_2}, F_1 \lor F_2, \Pi, \Pi', \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda', M^{\overline{i_2}}}\,\sigma$$

and π' is

$$\dfrac{\dfrac{\begin{matrix}(\varphi_1) \\ F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda\end{matrix} \qquad \dfrac{\begin{matrix}(\varphi_2) \\ G^{i_1}, F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda, G^{\overline{i_1}}\end{matrix}}{M^{i_2}, F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda, M^{\overline{i_2}}}\,\sigma}{F_1 \lor F_2, M^{i_2}, \Pi, \Pi, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda, M^{\overline{i_2}}}\,\rho}{F_1 \lor F_2, M^{i_2}, \Pi, \Pi', \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda', M^{\overline{i_2}}}\,c_*$$

then $\pi \rhd_b^1 \pi'$.

3. π is

$$\dfrac{\dfrac{\dfrac{\begin{matrix}(\varphi_1) \\ F_1, \Pi, G^{i_1}, \Gamma_1 \vdash \Delta_1, \Lambda, G^{\overline{i_1}}\end{matrix} \qquad \begin{matrix}(\varphi_2) \\ F_2, \Pi, G^{i_1}, \Gamma_2 \vdash \Delta_2, \Lambda, G^{\overline{i_1}}\end{matrix}}{F_1 \lor F_2, \Pi, G^{i_1}, \Pi, G^{i_1}, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, G^{\overline{i_1}}, \Lambda, G^{\overline{i_1}}}\,\rho}{G^{i_1}, F_1 \lor F_2, \Pi, \Pi', \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda', G^{\overline{i_1}}}\,c_*}{M^{i_2}, F_1 \lor F_2, \Pi, \Pi', \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda', M^{\overline{i_2}}}\,\sigma$$

and π' is

$$\dfrac{\dfrac{\dfrac{\begin{matrix}(\varphi_1) \\ G^{i_1}, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda, G^{\overline{i_1}}\end{matrix}}{M^{i_2}, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda, M^{\overline{i_2}}}\,\sigma \qquad \dfrac{\begin{matrix}(\varphi_2) \\ G^{i_1}, F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda, G^{\overline{i_1}}\end{matrix}}{M^{i_2}, F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda, M^{\overline{i_2}}}\,\sigma}{F_1 \lor F_2, \Pi, M^{i_2}, \Pi, M^{i_2}, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, M^{\overline{i_2}}, \Lambda, M^{\overline{i_2}}}\,\rho}{F_1 \lor F_2, \Pi, \Pi', M^{i_2}, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda', M^{\overline{i_2}}}\,c_*$$

then $\pi \rhd_b^1 \pi'$.

4. π is

$$\cfrac{\cfrac{\cfrac{\begin{matrix}(\varphi_1)\\ F_1,\Pi,\Gamma_1 \vdash \Delta_1,\Lambda\end{matrix} \qquad \begin{matrix}(\varphi_2)\\ F_2,\Pi,\Gamma_2\vdash\Delta_2,\Lambda\end{matrix}}{F_1\lor F_2,\Pi,\Pi,\Gamma_1,\Gamma_2\vdash\Delta_1,\Delta_2,\Lambda,\Lambda}\,\rho}{F_1\lor F_2,\Pi,\Pi',\Gamma_1,\Gamma_2\vdash\Delta_1,\Delta_2,\Lambda,\Lambda'}\,c_*}{M^i,F_1\lor F_2,\Pi,\Pi',\Gamma_1,\Gamma_2\vdash\Delta_1,\Delta_2,\Lambda,\Lambda',M^{\bar i}}\,w_*$$

and π' is

$$\cfrac{\cfrac{\cfrac{\begin{matrix}(\varphi_1)\\ F_1,\Pi,\Gamma_1 \vdash \Delta_1,\Lambda\end{matrix}}{M^i,F_1,\Pi,\Gamma_1\vdash\Delta_1,\Lambda,M^{\bar i}}\,w_* \qquad \begin{matrix}(\varphi_2)\\ F_2,\Pi,\Gamma_2\vdash\Delta_2,\Lambda\end{matrix}}{F_1\lor F_2,M^i,\Pi,\Pi,\Gamma_1,\Gamma_2\vdash\Delta_1,\Delta_2,\Lambda,\Lambda,M^{\bar i}}\,\rho}{F_1\lor F_2,M^i,\Pi,\Pi',\Gamma_1,\Gamma_2\vdash\Delta_1,\Delta_2,\Lambda,\Lambda',M^{\bar i}}\,c_*$$

then $\pi \rhd_b^1 \pi'$.

5. π is

$$\cfrac{\cfrac{\cfrac{\begin{matrix}(\varphi_1)\\ F_1,\Pi,\Gamma_1,G^i,G^i \vdash \Delta_1,\Lambda,G^{\bar i},G^{\bar i}\end{matrix} \qquad \begin{matrix}(\varphi_2)\\ F_2,\Pi,\Gamma_2\vdash\Delta_2,\Lambda\end{matrix}}{F_1\lor F_2,\Pi,\Pi,\Gamma_1,\Gamma_2,G^i,G^i\vdash\Delta_1,\Delta_2,\Lambda,\Lambda,G^{\bar i},G^{\bar i}}\,\rho}{G^i,G^i,F_1\lor F_2,\Pi,\Pi',\Gamma_1,\Gamma_2\vdash\Delta_1,\Delta_2,\Lambda,\Lambda',G^{\bar i},G^{\bar i}}\,c_*}{G^i,F_1\lor F_2,\Pi,\Pi',\Gamma_1,\Gamma_2\vdash\Delta_1,\Delta_2,\Lambda,\Lambda',G^{\bar i}}\,\sigma$$

and π' is

$$\cfrac{\cfrac{\cfrac{\begin{matrix}(\varphi_1)\\ F_1,\Pi,\Gamma_1,G^i,G^i \vdash \Delta_1,\Lambda,G^{\bar i},G^{\bar i}\end{matrix}}{F_1,\Pi,\Gamma_1,G^i\vdash\Delta_1,\Lambda,G^{\bar i}}\,\sigma \qquad \begin{matrix}(\varphi_2)\\ F_2,\Pi,\Gamma_2\vdash\Delta_2,\Lambda\end{matrix}}{F_1\lor F_2,\Pi,\Pi,\Gamma_1,\Gamma_2,G^i\vdash\Delta_1,\Delta_2,\Lambda,\Lambda,G^{\bar i}}\,\rho}{G^i,F_1\lor F_2,\Pi,\Pi',\Gamma_1,\Gamma_2\vdash\Delta_1,\Delta_2,\Lambda,\Lambda',G^{\bar i}}\,c_*$$

then $\pi \rhd_b^1 \pi'$.

6. π is

$$\cfrac{\cfrac{\cfrac{\begin{matrix}(\varphi_1)\\ F_1,\Pi,\Gamma_1 \vdash \Delta_1,\Lambda\end{matrix} \qquad \begin{matrix}(\varphi_2)\\ F_2,\Pi,\Gamma_2,G^i,G^i\vdash\Delta_2,\Lambda,G^{\bar i},G^{\bar i}\end{matrix}}{F_1\lor F_2,\Pi,\Pi,\Gamma_1,\Gamma_2,G^i,G^i\vdash\Delta_1,\Delta_2,\Lambda,\Lambda,G^{\bar i},G^{\bar i}}\,\rho}{G^i,G^i,F_1\lor F_2,\Pi,\Pi',\Gamma_1,\Gamma_2\vdash\Delta_1,\Delta_2,\Lambda,\Lambda',G^{\bar i},G^{\bar i}}\,c_*}{G^i,F_1\lor F_2,\Pi,\Pi',\Gamma_1,\Gamma_2\vdash\Delta_1,\Delta_2,\Lambda,\Lambda',G^{\bar i}}\,\sigma$$

and π' is

$$
\dfrac{\begin{matrix}(\varphi_1)\\ F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda\end{matrix} \qquad \dfrac{\begin{matrix}(\varphi_2)\\ F_2, \Pi, \Gamma_2, G^i, G^i \vdash \Delta_2, \Lambda, G^{\bar{i}}, G^{\bar{i}}\end{matrix}}{F_2, \Pi, \Gamma_2, G^i \vdash \Delta_2, \Lambda, G^{\bar{i}}}\,\sigma}{\dfrac{F_1 \vee F_2, \Pi, \Pi, \Gamma_1, \Gamma_2, G^i \vdash \Delta_1, \Delta_2, \Lambda, \Lambda, G^{\bar{i}}}{G^i, F_1 \vee F_2, \Pi, \Pi', \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda', G^{\bar{i}}}\,c_*}\,\rho
$$

then $\pi \rhd_b^1 \pi'$.

7. π is

$$
\dfrac{\dfrac{\dfrac{\begin{matrix}(\varphi_1)\\ F_1, \Pi, \Gamma_1, G_1 \vdash \Delta_1, \Lambda\end{matrix} \qquad \begin{matrix}(\varphi_2)\\ F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda\end{matrix}}{F_1 \vee F_2, \Pi, \Pi, \Gamma_1, G_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda}\,\rho}{G_1, F_1 \vee F_2, \Pi, \Pi', \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda'}\,c_* \qquad \begin{matrix}(\varphi_3)\\ G_2, \Sigma \vdash \Theta\end{matrix}}{G_1 \vee G_2, F_1 \vee F_2, \Pi, \Pi', \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda, \Lambda'}\,\sigma
$$

and π' is

$$
\dfrac{\dfrac{\dfrac{\begin{matrix}(\varphi_1)\\ G_1, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda\end{matrix} \qquad \begin{matrix}(\varphi_3)\\ G_2, \Sigma \vdash \Theta\end{matrix}}{G_1 \vee G_2, F_1, \Pi, \Gamma_1, \Sigma \vdash \Theta, \Delta_1, \Lambda}\,\sigma \qquad \begin{matrix}(\varphi_2)\\ F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda\end{matrix}}{F_1 \vee F_2, G_1 \vee G_2, \Pi, \Pi, \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda, \Lambda}\,\rho}{F_1 \vee F_2, G_1 \vee G_2, \Pi, \Pi', \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda, \Lambda'}\,c_*
$$

then $\pi \rhd_b^1 \pi'$.

8. π is

$$
\dfrac{\dfrac{\dfrac{\begin{matrix}(\varphi_1)\\ F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda\end{matrix} \qquad \begin{matrix}(\varphi_1)\\ F_2, \Pi, \Gamma_2, G_1 \vdash \Delta_2, \Lambda\end{matrix}}{F_1 \vee F_2, \Pi, \Pi, \Gamma_1, \Gamma_2, G_1 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda}\,\rho}{G_1, F_1 \vee F_2, \Pi, \Pi', \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda'}\,c_* \qquad \begin{matrix}(\varphi_1)\\ G_2, \Sigma \vdash \Theta\end{matrix}}{G_1 \vee G_2, F_1 \vee F_2, \Pi, \Pi', \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda, \Lambda'}\,\sigma
$$

and π' is

$$
\dfrac{\dfrac{\begin{matrix}(\varphi_1)\\ F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda\end{matrix} \qquad \dfrac{\begin{matrix}(\varphi_2)\\ G_1, F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda\end{matrix} \qquad \begin{matrix}(\varphi_3)\\ G_2, \Sigma \vdash \Theta\end{matrix}}{G_1 \vee G_2, F_2, \Pi, \Gamma_2, \Sigma \vdash \Theta, \Delta_2, \Lambda}\,\sigma}{F_1 \vee F_2, G_1 \vee G_2, \Pi, \Pi, \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda, \Lambda}\,\rho}{F_1 \vee F_2, G_1 \vee G_2, \Pi, \Pi', \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda, \Lambda'}\,c_*
$$

then $\pi \rhd_b^1 \pi'$.

9. π is

$$\dfrac{\dfrac{\dfrac{\dfrac{(\varphi_1)}{F_1, \Pi, G_1, \Gamma_1 \vdash \Delta_1, \Lambda} \quad \dfrac{(\varphi_2)}{F_2, \Pi, , G_1 \Gamma_2 \vdash \Delta_2, \Lambda}}{F_1 \vee F_2, \Pi, G_1, \Pi, G_1, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda}\,\rho}{F_1 \vee F_2, \Pi, \Pi', G_1, G_1', \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda'}\,c_* \quad \dfrac{(\varphi_3)}{G_2, \Sigma \vdash \Theta}}{G_1 \vee G_2, G_1', F_1 \vee F_2, \Pi, \Pi', \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda, \Lambda'}\,\sigma$$

and π' is

$$\dfrac{\dfrac{\dfrac{\dfrac{(\varphi_1)}{G_1, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda} \quad \dfrac{(\varphi_3)}{G_2, \Sigma \vdash \Theta}}{G_1 \vee G_2, F_1, \Pi, \Gamma_1, \Sigma \vdash \Theta, \Delta_1, \Lambda}\,\sigma \quad (\phi)}{F_1 \vee F_2, \Pi, G_1 \vee G_2, \Pi, G_1 \vee G_2, \Gamma_1, \Gamma_2, \Sigma, \Sigma \vdash \Theta, \Theta, \Delta_1, \Delta_2, \Lambda, \Lambda}\,\rho}{F_1 \vee F_2, G_1 \vee G_2, (G_1 \vee G_2)', \Pi, \Pi', \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Theta', \Delta_1, \Delta_2, \Lambda, \Lambda'}\,c_*$$

where ϕ is

$$\dfrac{\dfrac{(\varphi_2)}{G_1, F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda} \quad \dfrac{(\varphi_3)}{G_2, \Sigma \vdash \Theta}}{G_1 \vee G_2, F_2, \Pi, \Gamma_2, \Sigma \vdash \Theta, \Delta_2, \Lambda}\,\sigma$$

then $\pi \rhd_b^1 \pi'$.

Note here that the inference c_* is used as follows: If in a sequent Π, Π occurs, then after the application of c_* we get Π, Π' with Π' possibly empty or equal to Π. Hence, c_* contracts either one of the occurrences of Π, or none of them.

The $\rhd_b$ relation is defined as the transitive and reflexive closure of the compatible closure of the $\rhd_b^1$ relation.

7.4.1 Proof length

Consider an $\mathbf{LK}_{sk}$-proof π and the corresponding **LK**-proof φ where $\pi \rhd_{u/b} \varphi$ holds. We will analyse how the transformation $\rhd_{u/b}$ influences the proof-length $l(\varphi)$ (see Definition 2.32) of the **LK**-proof φ with respect to the proof-length $l(\pi)$ of the $\mathbf{LK}_{sk}$-proof π.

Looking at the permutations of Definition 7.4.1 and Definition 7.4.2 we see that $l(\pi) = l(\pi')$ for all permutations except for $\rhd_b$ case 3 and case 9 in Definition 7.4.2. For the sake of simplicity, we introduce two new definitions:

Definition 7.4.3. $\Pi_=$ is the set of permutations where the proof-length does not change, i.e. where $l(\pi) = l(\pi')$ holds.

Definition 7.4.4. $\Pi_<$ is the set of permutations where the proof-length differs, i.e. where $l(\pi) < l(\pi')$ holds.

Note that case 3 and case 9 of Definition 7.4.2 are the only permutations $\in \Pi_<$ and all other permutations of Definition 7.4.1 and Definition 7.4.2 belong to the set $\Pi_=$, so there is no single case where $l(\pi) > l(\pi')$ holds.

If the transformation of a $\mathbf{LK}_{sk}$-proof π into a $\mathbf{LK}$-proof φ uses only permutations $\in \Pi_=$ we clearly have that $l(\pi) = l(\varphi)$. However, if also permutations $\in \Pi_<$ occur in the transformation, we observe the following:

Consider case 9 from Definition 7.4.2:
π is

$$\cfrac{\cfrac{\cfrac{\begin{matrix}\varphi_1 \\ F_1, \Pi, G_1, \Gamma_1 \vdash \Delta_1, \Lambda\end{matrix} \quad \begin{matrix}\varphi_2 \\ F_2, \Pi, , G_1 \Gamma_2 \vdash \Delta_2, \Lambda\end{matrix}}{F_1 \vee F_2, \Pi, G_1, \Pi, G_1, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda}\,\rho}{F_1 \vee F_2, \Pi, \Pi', G_1, G_1', \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Lambda, \Lambda'}\,c_* \quad \begin{matrix}\varphi_3 \\ G_2, \Sigma \vdash \Theta\end{matrix}}{G_1 \vee G_2, G_1', F_1 \vee F_2, \Pi, \Pi', \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, \Lambda, \Lambda'}\,\sigma$$

and π' is

$$\cfrac{\cfrac{\cfrac{\begin{matrix}\varphi_1 \\ G_1, F_1, \Pi, \Gamma_1 \vdash \Delta_1, \Lambda\end{matrix} \quad \begin{matrix}\varphi_3 \\ G_2, \Sigma \vdash \Theta\end{matrix}}{G_1 \vee G_2, F_1, \Pi, \Gamma_1, \Sigma \vdash \Theta, \Delta_1, \Lambda}\,\sigma \quad (\phi)}{F_1 \vee F_2, \Pi, G_1 \vee G_2, \Pi, G_1 \vee G_2, \Gamma_1, \Gamma_2, \Sigma, \Sigma \vdash \Theta, \Theta, \Delta_1, \Delta_2, \Lambda, \Lambda}\,\rho}{F_1 \vee F_2, G_1 \vee G_2, (G_1 \vee G_2)', \Pi, \Pi', \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Theta', \Delta_1, \Delta_2, \Lambda, \Lambda'}\,c_*$$

where ϕ is

$$\cfrac{\begin{matrix}\varphi_2 \\ G_1, F_2, \Pi, \Gamma_2 \vdash \Delta_2, \Lambda\end{matrix} \quad \begin{matrix}\varphi_3 \\ G_2, \Sigma \vdash \Theta\end{matrix}}{G_1 \vee G_2, F_2, \Pi, \Gamma_2, \Sigma \vdash \Theta, \Delta_2, \Lambda}\,\sigma$$

then $\pi \rhd_b^1 \pi'$.
In the transformation from π to π' the subproof φ_3 is duplicated. This means, that if the subproof φ_3 contains sufficiently many subproofs of the same kind as in case 9, the complexity of the transformation will be exponential in the size of the input proof. To clarify the meaning of sufficiently many subproofs we introduce the next definition.

Definition 7.4.5. Sequence of proofs of same kind. We define a sequence of proofs $(\varphi_n)_{n\in\mathbb{N}}$ of proofs of the same kind as in Definition 7.4.2 case 9 as follows:

Let φ_1 be the following proof:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{F_1, G_1(t), \Gamma_1 \vdash \Delta_1, F_1, G_1(t)}{F_1, G_1(t), \Gamma_1 \vdash \Delta_1, F_1, (\forall x)G_1(x)}\forall_r \quad \dfrac{F_2, G_1(t), \Gamma_2 \vdash \Delta_2, F_2, G_1(t)}{F_2, G_1(t), \Gamma_2 \vdash \Delta_2, F_2, (\forall x)G_1(x)}\forall_r}{F_1 \lor F_2, G_1(t), G_1(t), \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, F_1, (\forall x)G_1(x), F_2, (\forall x)G_1(x)}\lor_l}{F_1 \lor F_2, G_1(t), \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, F_1, (\forall x)G_1(x), F_2}c_* \quad G_2(t), \Sigma \vdash \Theta, G_2(t)}{G_1(t) \lor G_2(t), F_1 \lor F_2, \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, F_1, (\forall x)G_1(x), F_2, (\forall x)G_2(x)}\lor_l}{(\forall x)(G_1(x) \lor G_2(x)), F_1 \lor F_2, \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, F_1, (\forall x)G_1(x), F_2, (\forall x)G_2(x)}\forall_l$$

And let φ_n for $n > 1$ be

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{F_1, G_1(t), \Gamma_1 \vdash \Delta_1, F_1, G_1(t)}{F_1, G_1(t), \Gamma_1 \vdash \Delta_1, F_1, (\forall x)G_1(x)}\forall_r \quad \dfrac{F_2, G_1(t), \Gamma_2 \vdash \Delta_2, F_2, G_1(t)}{F_2, G_1(t), \Gamma_2 \vdash \Delta_2, F_2, (\forall x)G_1(x)}\forall_r}{F_1 \lor F_2, G_1(t), G_1(t), \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, F_1, (\forall x)G_1(x), F_2, (\forall x)G_1(x)}\lor_l}{F_1 \lor F_2, G_1(t), \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, F_1, (\forall x)G_1(x), F_2}c_* \quad \begin{array}{c}\varphi_{n-1}\\ G_2(t), \Sigma' \vdash \Theta', G_2(t)\end{array}}{G_1(t) \lor G_2(t), F_1 \lor F_2, \Gamma_1, \Gamma_2, \Sigma' \vdash \Theta', \Delta_1, \Delta_2, F_1, (\forall x)G_1(x), F_2, (\forall x)G_2(x)}\lor_l}{(\forall x)(G_1(x) \lor G_2(x)), F_1 \lor F_2, \Gamma_1, \Gamma_2, \Sigma' \vdash \Theta', \Delta_1, \Delta_2, F_1, (\forall x)G_1(x), F_2, (\forall x)G_2(x)}\forall_l$$

where Σ' in φ_n is defined to be the end-sequent of φ_{n-1} except that the subformula Σ in the end-sequent of φ_{n-1} is replaced by $\Sigma \backslash \{G_2\}$ and Θ' in φ_n is defined to be the end-sequent of φ_{n-1} except that the subformula Θ' in the end-sequent of φ_{n-1} is replaced by $\Theta \backslash \{G_2\}$. Note that for simplicity we do not use labels in the proofs above.

Now consider a sequence of proofs $(\varphi_n)_{n\in\mathbb{N}}$ as defined in Definition 7.4.5. We will show that for this sequence, the complexity of the transformation (from $\mathbf{LK}_{sk}$-proofs into **LK**-proofs) is exponential in the size of the input proofs.

Theorem 7.4.1. *Complexity of transformations. There exists a sequence of* $\boldsymbol{LK}_{sk}$*-proofs* $(\varphi_n)_{n\in\mathbb{N}}$ *s.t.*

$$l(\varphi_n) \leq k * n,$$

*and after the transformation into a sequence of **LK**-proofs* $(\varphi_n^*)_{n\in\mathbb{N}}$ *we get*

$$l(\varphi_n^*) \geq 2^{l*n}$$

Proof. Let $(\varphi_n^*)_{n\in\mathbb{N}}$ be the sequence of proofs defined in Definition 7.4.5. Note, the strong quantifiers $\forall_r$ are at the top of φ_n, the weak ones at the bottom. To establish the validity of eigenvariable conditions in the final **LK**-proof the full transformation has to be performed. First we show that $l(\varphi_n) \leq k * n$ holds for $k = 9$. First note

$$l(\varphi_1) = 9$$

$$l(\varphi_2) = 9 + 8$$

$$l(\varphi_3) = 9 + 8 + 8$$

$$\vdots$$

$$l(\varphi_n) = 9 + (n-1) * 8$$

The proof is by induction on n.
Base case $n = 1$:

$$l(\varphi_1) = 9 \leq 9 * 1$$

Now assume as induction hypothesis (IH) that $l(\varphi_m) \leq 9 * m$ holds for for all $m \leq n$. Consider $m = n + 1$. Since $l(\varphi_n) = l(\varphi_{n-1}) + 8$ for all n we get

$$l(\varphi_{n+1}) = l(\varphi_n) + 8.$$

By (IH) we know that $l(\varphi_n) \leq 9 * n$, hence

$$l(\varphi_n) + 8 \leq 9 * n + 8$$

Since

$$9 * n + 8 < 9 * n + 9 = 9 * (n + 1)$$

we conclude that

$$l(\varphi_{n+1}) \leq 9 * (n + 1).$$

Therefore we obtain $l(\varphi_n) \leq k * n$ for $k = 9$.

Now we show that $l(\varphi_n^*) \geq 2^{l*n}$ for some $l \in \mathbb{N}$. φ_1^* is the transformed proof φ_1:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{F_1, G_1(t), \Gamma_1 \vdash \Delta_1, F1, G_1(t) \qquad G_2(t), \Sigma \vdash \Theta, G_2(t)}{G_1(t) \vee G_2(t), F_1, \Gamma_1, \Sigma \vdash \Theta, \Delta_1, F1, G_1(t), G_2(t)}\vee_l}{(\forall x)(G_1(x) \vee G_2(x)), F_1, \Gamma_1, \Sigma \vdash \Theta, \Delta_1, F1, G_1(t), G_2(t)}\forall_l}{G_1(t) \vee G_2(t), F_1, \Gamma_1, \Sigma \vdash \Theta, \Delta_1, F1, (\forall x)G_1(x), (\forall x)G_2(x)}\forall_r \qquad \phi}{F_1 \vee F_2, (\forall x)(G_1(x) \vee G_2(x)), G_1 \vee G_2, \Gamma_1, \Gamma_2, \Sigma, \Sigma \vdash \Theta, \Theta, \Delta_1, \Delta_2, F1, (\forall x)G_1(x), (\forall x)G_2(x), (\forall x)G_1(x), F_2, (\forall x)G_2(x)}\vee_l}{F_1 \vee F_2, (\forall x)(G_1(x) \vee G_2(x)), \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, F1, (\forall x)G_1(x), (\forall x)G_2(x), F_2}c_*$$

where ϕ is

$$\dfrac{\dfrac{\dfrac{G_1(t), F_2, \Gamma_2 \vdash \Delta_2, G_1(t), F_2 \qquad G_2(t), \Sigma \vdash \Theta, G_2(t)}{G_1(t) \vee G_2(t), F_2, \Gamma_2, \Sigma \vdash \Theta, \Delta_2, G_1(t), F_2, G_2(t)}\vee_l}{(\forall x)(G_1(x) \vee G_2(x)), F_2, \Gamma_2, \Sigma \vdash \Theta, \Delta_2, G_1(t), F_2, G_2(t)}\forall_l}{(\forall x)(G_1(x) \vee G_2(x)), F_2, \Gamma_2, \Sigma \vdash \Theta, \Delta_2, (\forall x)G_1(x), F_2, (\forall x)G_2(x)}\forall_r$$

and φ_n^* is

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{F_1, G_1(t), \Gamma_1 \vdash \Delta_1, F1, G_1(t) \qquad G_2(t), \Sigma \vdash \Theta, G_2(t)}{G_1(t) \vee G_2(t), F_1, \Gamma_1, \Sigma \vdash \Theta, \Delta_1, F1, G_1(t), G_2(t)}\vee_l}{(\forall x)(G_1(x) \vee G_2(x)), F_1, \Gamma_1, \Sigma \vdash \Theta, \Delta_1, F1, G_1(t), G_2(t)}\forall_l}{G_1(t) \vee G_2(t), F_1, \Gamma_1, \Sigma \vdash \Theta, \Delta_1, F1, (\forall x)G_1(x), (\forall x)G_2(x)}\forall_r \qquad \overset{(\varphi^*_{n-1})}{\phi}}{F_1 \vee F_2, (\forall x)(G_1(x) \vee G_2(x)), G_1 \vee G_2, \Gamma_1, \Gamma_2, \Sigma, \Sigma \vdash \Theta, \Theta, \Delta_1, \Delta_2, F1, (\forall x)G_1(x), (\forall x)G_2(x), (\forall x)G_1(x), F_2, (\forall x)G_2(x)}\vee_l}{F_1 \vee F_2, (\forall x)(G_1(x) \vee G_2(x)), \Gamma_1, \Gamma_2, \Sigma \vdash \Theta, \Delta_1, \Delta_2, F1, (\forall x)G_1(x), (\forall x)G_2(x), F_2}c_*$$

where ϕ is

$$\dfrac{\dfrac{\dfrac{G_1(t), F_2, \Gamma_2 \vdash \Delta_2, G_1(t), F_2 \qquad \overset{(\varphi^*_{n-1})}{G_2(t), \Sigma \vdash \Theta, G_2(t)}}{G_1(t) \vee G_2(t), F_2, \Gamma_2, \Sigma \vdash \Theta, \Delta_2, G_1(t), F_2, G_2(t)}\ \vee_l}{(\forall x)(G_1(x) \vee G_2(x)), F_2, \Gamma_2, \Sigma \vdash \Theta, \Delta_2, G_1(t), F_2, G_2(t)}\ \forall_l}{(\forall x)(G_1(x) \vee G_2(x)), F_2, \Gamma_2, \Sigma \vdash \Theta, \Delta_2, (\forall x)G_1(x), F_2, (\forall x)G_2(x)}\ \forall_r$$

First note

$$l(\varphi^*_1) = 12$$

$$l(\varphi^*_2) = 10 + 2 * 12 = 34$$

$$l(\varphi^*_3) = 10 + 2 * 34 = 78$$

$$\vdots$$

A careful analysis of the structure of $l(\varphi^*_n)$ shows that

$$l(\varphi^*_n) = \begin{cases} 12 & \text{if } n = 1 \\ 10 + 2 * l(\varphi^*_{n-1}) & \text{otherwise} \end{cases}$$

Note that for $n = 1$ we have

$$l(\varphi^*_1) = 12 \geq 2^1.$$

It can be proved by induction that for $n > 1$ it holds that $l(\varphi^*_n) = (2^{n-1} - 1) * 10 + 2^{n-1} * l(\varphi^*_1)$. Therefore

$$\begin{aligned} l(\varphi^*_n) &= (2^{n-1} - 1) * 10 + 2^{n-1} * l(\varphi^*_1) \\ &= 2^{n-1} * 10 - 10 + 2^{n-1} * 12 \\ &= 2^{n-1} * 22 - 10 \end{aligned}$$

Since

$$2^{n-1} * 22 - 10 = 2^{n-1} * 2 * 11 - 10 = 2^n * 11 - 10$$

we finally obtain

$$l(\varphi^*_n) \geq 2^n.$$

Hence, we have proved that $l(\varphi^*_n) \geq 2^{l*n}$ holds for $l = 1$. □

CHAPTER 8

Conclusion

The goal of this thesis was to define a Skolem-free CERES-method for first-order logic and to speed-up the Herbrand sequent extraction.

In Section 7.1 we have defined a Skolem-free CERES-method, where we used the cut-free calculus $\mathbf{LK}_{sk}$ introduced in [13] and [22]. In this calculus we use Skolem terms instead of the eigenvariables of **LK** and therefore do not have to take care of the eigenvariable conditions.

We construct the characteristic sequent set and the projections, as described in [13] and [22]. Using the resolution calculus for the original CERES-method, [6] and [3], we construct an **LK**-refutation of the reduct of the characteristic sequent set.

The Herbrand sequents are usually extracted from the ACNF, which we get after CERES was applied to a proof containing cuts. Therefore, our method would need a transformation into **LK**. In [13] and [22] it was shown that $\mathbf{LK}_{sk}$-proofs can be transformed to **LK**-proofs. However, we skip this transformation and show that Herband sequents can be extracted in an earlier step in CERES. Indeed, it is possible to extract Herbrand sequents from the resolution refutation and the corresponding projections, which we showed in Section 7.2.

In Section 7.3 we proved that we do not need the transformation from $\mathbf{LK}_{sk}$-proofs into **LK**-proofs to be able to extract Herbrand sequents, because the sequents relevant for the extraction are not modified during the transformation, which was originally used to show soundness of $\mathbf{LK}_{sk}$. Hence, the whole transformation is obsolete which lead us to the investigation of the complexity of the transformation.

In Section 7.4 we proved that the complexity of the transformation can be exponential in the size of the input proof, which means that we defined a new method for Herbrand sequent extraction, which is faster than the one defined by the original Skolem-free CERES-method.

Since we proved that with the Skolem-free CERES-method the Herbrand sequents can be extracted more efficiently, an implementation of this method could follow in future work.

Moreover, a generalization to higher-order logic (resulting in an efficient computation of expansion trees) would be of major importance in mathematical proof mining.

Bibliography

[1] P. B. Andrews. Resolution in type theory. *Journal of Symbolic Logic*, pages 36(3):414–432, 1971.

[2] M. Baaz, A. Ciabattoni, and C. G. Fermüller. Cut-elimination for first-order Gödel logic by hyperclause resolution. In *Logic for Programming, Artificial Intelligence, and Reasoning*, pages 451–466. Springer, 2008.

[3] M. Baaz and A. Leitsch. Cut-elimination and redundancy-elimination by resolution. *Journal of Symbolic Computation*, 29(2):149–177, 2000.

[4] M. Baaz and A. Leitsch. Ceres in many-valued logics. In *Logic for Programming, Artificial Intelligence, and Reasoning*, pages 1–20. Springer, 2005.

[5] M. Baaz and A. Leitsch. Towards a clausal analysis of cut-elimination. *Journal of Symbolic Computation*, 41(3-4):381–410, 2006.

[6] M. Baaz and A. Leitsch. *Methods of Cut-elimination*, volume 34. Springer, 2011.

[7] A. Church. A formulation of the simple theory of types. *Journal of Symbolic Logic*, pages 5(2):56–68, 1940.

[8] H. Fürstenberg and B. Weiss. Topological dynamics and combinatorial number theory. *Journal d'Analyse Mathematique*, pages 34(1):61–85, 1978.

[9] G. Gentzen. Untersuchungen über das logische Schließen. *Mathematische Zeitschrift*, 39:176–210,405–431, 1934-35.

[10] P. Gerhardy and U. Kohlenbach. Extracting Herbrand disjunctions by functional interpretation. *Archive for Mathematical Logic*, pages 44(5):633–644, 2005.

[11] J. Y. Girard. *Proof Theory and Logical Complexity*, volume 1. Biblopolis, Napoli,1987.

[12] K. Gödel. über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes. *Dialectica*, 1958.

[13] S. Hetzl, A. Leitsch, and D. Weller. Ceres in higher-order logic. *Annals of Pure and Applied Logic*, 162(12):1001–1034, 2011.

[14] A. Leitsch. *The resolution calculus.* Springer-Verlag New York, Inc., 1997.

[15] A. Leitsch. *On proof mining by cut-elimination.* Invited talk, July 18, All about Proofs, Proofs for all (APPA), Vienna Summer of Logic, Vienna, Austria, 2014.

[16] G. Reis. *Cut-elimination by resolution in intuitionistic logic.* PhD thesis, Vienna University of Technology, 2014.

[17] J. A. Robinson. A machine-oriented logic based on the resolution principle. *Journal of the ACM (JACM)*, 1965.

[18] K. Schütte. *Beweistheorie.* Springer, 1960.

[19] W. W. Tait. Normal derivability in classical logic. In *The Syntax and Semantics of Infinitary Languages*, pages 204–236. Springer, 1968.

[20] G. Takeuti. *Proof Theory.* Courier Dover Publications, 2013.

[21] B. L. Van der Waerden. Beweis einer Baudetschen Vermutung. *Nieuw Archiv Wiskunde*, pages 15(2):212–216, 1927.

[22] D. Weller. *CERES in Higher-Order Logic.* PhD thesis, Vienna University of Technology, 2010.

[23] B. Woltzenlogel Paleo. *A General Analysis of Cut-Elimination by CERes.* PhD thesis, Vienna University of Technology, 2009.