On proof mining by cut-elimination

Alexander Leitsch

Vienna University of Technology
leitsch@logic.at

Abstract. We present cut-elimination as a method of proof mining, in the sense that hidden mathematical information can be extracted by eliminating lemmas from proofs. We present reductive methods for cut-elimination and the method `ceres` (cut-elimination by resolution). A comparison of `ceres` with reductive methods is given and it is shown that the asymptotic behavior of `ceres` is superior to that of reductive methods (nonelementary speed-up). It is illustrated, how `ceres` can be extended and applied in practice for analyzing mathematical proofs. Finally we we give an application of `ceres` to a well-known proof of the infinitude of primes by Fürstenberg; this proof uses topological lemmas based on arithmetic progressions. These topological lemmas of the proof are eliminated by `ceres` and Euclid's construction of primes is extracted. We also touch the problem of cut-elimination by resolution on induction proofs and discuss the limits of the method.

1 Introduction

What is a mathematical proof? Just a *verification* of a statement or a key to *understand* a theorem? One and the same theorem may have several, possibly very different mathematical proofs, and each of them contains a specific form of *mathematical information.*

Mathematics in general is based on the structuring of reasoning by intermediate statements, the *lemmas*: this strongly increases the efficiency of mathematical thinking as the mathematician is not forced to have a proof of a lemma in mind when he makes use of it. He even might not know any proof of the lemma, but simply trusts other mathematicians concerning its truth.

The drawback of the use of lemmas is, however, that only their truth but not their proofs are reflected in the derivations of their end-statements. One of the most important insights in mathematical logic is Gentzen's Hauptsatz [16]. It states that lemmas (cuts) can be algorithmically eliminated from given first-order derivations. The result is a streamlined lemma-free proof combining all subproofs of the original derivation: the cut-free derivation. Gentzen's groundbreaking result has been motivated by Hilbert's distinction of *ideal* and *real* objects in mathematics, where the lemmas are supposed to encode properties of ideal objects.

The removal of cuts corresponds to the elimination of intermediate statements (lemmas) from proofs resulting in a proof which is *analytic* in the sense, that

all statements in the proof are subformulas of the result. Therefore, the proof of a combinatorial statement (possibly using theories outside the theory of the statement itself) is converted into a purely combinatorial proof.

While Gentzen's cut-elimination theorem found its immediate applications in abstract proof theory (in particular in proving consistency results), the technique of cut-elimination turned out to be fruitful in the analysis of "real" mathematical proofs. A famous application of cut-elimination to mathematical proofs is Girard's analysis of a topological proof of van der Waerden's theorem (Given a partition of $\mathbb{N} = C_1 \cup \cdots\cdots \cup C_k$, one of the sets C_i contains arbitrarily long arithmetic progressions) [17]. In a formal sense Girard's analysis of van der Waerden's theorem is the application of cut-elimination to the topological proof of Fürstenberg/Weiss with the "perspective" of obtaining van der Waerden's (combinatorial) proof. Naturally, an application of a complex proof transformation like cut-elimination by humans requires a goal oriented strategy.

The development of the method `ceres` [7] (cut-elimination by resolution) was inspired by the idea to fully automate cut-elimination on real mathematical proofs, with the aim of obtaining new interesting elementary proofs. While a fully automated treatment proved successful for mathematical proofs of moderate complexity (e.g. the tape proof [3] and the lattice proof [20], more complex mathematical proofs required an interactive use of `ceres`; this way we successfully analyzed Füstenberg's proof of the infinitude of primes (see [4] and [1]) and obtained Euclid's argument of prime construction. This proof, though much simpler than the proof of the Fürstenberg/Weiss proof of van der Waerden's theorem, is sufficiently "complex" in the sense that it proves a number theoretic result by a topological argument. By the use of `ceres` the topological proof was transformed into a number theoretic one, namely to that of Euclid. Though the analysis by `ceres` could not be fully automated, even its interactive use proved to be superior to the reductive cut-elimination method (based on Gentzen's proof) due to additional structural information given by the characteristic clause set (to be defined in Section 3).

`ceres` [7, 9] is a cut-elimination method that is based on resolution. The method roughly works as follows: The structure of the proof containing cuts is encoded in an unsatisfiable set of clauses $\mathcal{C}$ (the *characteristic clause set*). A resolution refutation γ of $\mathcal{C}$, which is obtained using a first-order theorem prover, serves as a skeleton for an atomic cut normal form ψ, a new proof which contains at most atomic cuts. γ is transformed to ψ by replacing the leaves of γ by so-called *proof projections*, which are essentially cut-free parts of the original proof. This method of cut-elimination has been implemented in the system `ceres`[1]. The system is capable of dealing with formal proofs in an extended version of **LK**, among them also very large ones.

Cut-elimination is not the only tool to *mine* proofs. An alternative method, the extraction of functionals, is based on Gödel's dialectica interpretation [18] and allows the construction of programs from proofs (see [11] and [12] for applications

[1] available at `http://www.logic.at/ceres/`

to mathematical proofs). Not only the result of the functional extraction method is different, also its range of applicability. Its advantage is the handling of the induction rule, which poses serious problems to cut-elimination; its disadvantage is the restriction to proofs of Π_2-statements (statements of the form $\forall x.\exists y.A(x,y)$ for A quantifier-free), while cut-elimination can be applied to arbitrary statements. Both methods have in common that they reveal hidden structures in proofs and provide new mathematical information by proof-transformation.

2 A Proof System for Cut-Elimination

As a basis for our investigations we use the sequent calculus **LK** (defined by Gerhard Gentzen[16]). In our version of **LK** we do not use an exchange rule as our sequents are based on the multi-set structure. There are several extensions of **LK** which are useful for analyzing mathematical proofs; we will mention some of them in Section 4.

Definition 1. *Let $\mathcal{A}$ and $\mathcal{B}$ be two multi-sets of formulas and $\vdash$ be a symbol not belonging to the logical language. Then $\mathcal{A} \vdash \mathcal{B}$ is called a sequent.*

Definition 2. *Let $S\colon A_1, \ldots, A_n \vdash B_1, \ldots, B_m$ be a sequent and $\mathcal{M}$ be an interpretation over the signature of $\{A_1, \ldots, A_n, B_1, \ldots B_m\}$. Then S is valid in $\mathcal{M}$ if the formula $(A_1 \wedge \cdots \wedge A_n) \rightarrow (B_1 \vee \cdots \vee B_m)$ is valid in $\mathcal{M}$. S is called* valid *if S is valid in all interpretations.*

Definition 3 (LK). *As axioms of the calculus we take the sequents $A \vdash A$ for atomic formulas A.*

There are two groups of rules, the logical and the structural ones. All rules with the exception of the cut rule have left and right versions; left versions are denoted by $\xi : l$, right versions by $\xi : r$. A and B denote formulas, $\Gamma, \Delta, \Pi, \Lambda$ multi-sets of formulas

The logical rules:

- *$\wedge$-introduction:*

$$\frac{A, \Gamma \vdash \Delta}{A \wedge B, \Gamma \vdash \Delta} \wedge\colon l_1 \qquad \frac{B, \Gamma \vdash \Delta}{A \wedge B, \Gamma \vdash \Delta} \wedge\colon l_2 \qquad \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash \Delta, A \wedge B} \wedge\colon r$$

- *$\vee$-introduction:*

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

- *$\rightarrow$-introduction:*

$$\frac{\Gamma \vdash \Delta, A \quad B, \Pi \vdash \Lambda}{A \rightarrow B, \Gamma, \Pi \vdash \Delta, \Lambda} \rightarrow\colon l \qquad \frac{A, \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \rightarrow B} \rightarrow\colon r$$

– $\neg$-*introduction:*

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

– $\forall$-*introduction:* [2] [3]

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

– *The logical rules for* $\exists$*-introduction (the variable conditions for* $\exists : l$ *are these for* $\forall : r$*, and similarly for* $\exists : r$ *and* $\forall : l$*):*

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

The structural rules:

– weakening*:*

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

– contraction:

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

– *The* cut rule:

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

LK is particularly suited for proof analysis, as the cut-rule (representing the use of lemmas) can be constructively eliminated from proofs; the resulting proof is analytic, i.e. the whole material of the proof consists of subformulas of the end-sequent. In typical Hilbert-type calculi the only rules are modus ponens and the generalization rule. There is no way to eliminate modus ponens from a typical Hilbert type calculus, and thus the elimination of lemmas cannot be described in such a framework. As we work in classical logic, natural deduction (which has introduction and elimination rules and is closer to sequent calculus), being basically a calculus for intuitionistic logic, is also not the optimal choice.

Example 1. We give two proofs of the same sequent, one with cut, the other without it.

Let $\varphi =$

$$\dfrac{\dfrac{\dfrac{\dfrac{P(a) \vdash P(a)}{P(a) \vdash P(a) \vee Q(a)}\ \vee : r_1}{P(a) \vdash \exists y(P(y) \vee Q(y))}\ \exists : r \qquad \dfrac{\dfrac{Q(b) \vdash Q(b)}{Q(b) \vdash P(b) \vee Q(b)}\ \vee : r_2}{Q(b) \vdash \exists y(P(y) \vee Q(y))}\ \exists : r}{P(a) \vee Q(b) \vdash \exists y(P(y) \vee Q(y))}\ \vee : l \qquad \dfrac{\exists y(P(y) \vee Q(y)), \forall x.\neg P(x) \vdash \exists z.Q(z)}{(\chi)}}{P(a) \vee Q(b), \forall x.\neg P(x) \vdash \exists z.Q(z)}\ cut$$

[2] t is an arbitrary term containing only free variables.

[3] y is a free variable which may not occur in Γ, Δ. y is called an eigenvariable.

for $\chi =$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(\alpha) \vdash P(\alpha)}{P(\alpha), \neg P(\alpha) \vdash}\, \neg\colon l}{P(\alpha), \neg P(\alpha) \vdash Q(\alpha)}\, w\colon r \quad \dfrac{Q(\alpha) \vdash Q(\alpha)}{Q(\alpha), \neg P(\alpha) \vdash Q(\alpha)}\, w\colon l}{P(\alpha) \vee Q(\alpha), \neg P(\alpha) \vdash Q(\alpha)}\, \vee\colon l}{P(\alpha) \vee Q(\alpha), \neg P(\alpha) \vdash \exists z.Q(z)}\, \exists\colon r}{P(\alpha) \vee Q(\alpha), \forall x.\neg P(x) \vdash \exists z.Q(z)}\, \forall\colon l}{\exists y(P(y) \vee Q(y)), \forall x.\neg P(x) \vdash \exists z.Q(z)}\, \exists\colon l$$

where α is an eigenvariable. When we search for a witness for the z in $\exists z.Q(z)$ and trace the proof part χ via the ancestors of $\exists z.Q(z)$ we see that no direct answer can be obtained. In fact we can trace α until it "disappears" by the $\exists\colon l$-rule.The following cut-free proof, which can be obtained via Gentzen-type cut-elimination, provides more information about z:

$\psi =$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(a) \vdash P(a)}{P(a), \neg P(a) \vdash}\, \neg\colon l}{P(a), \neg P(a) \vdash Q(b)}\, w\colon r \quad \dfrac{Q(b) \vdash Q(b)}{Q(b), \neg P(a) \vdash Q(b)}\, w\colon l}{P(a) \vee Q(b), \neg P(a) \vdash Q(b)}\, \vee\colon l}{P(a) \vee Q(b), \neg P(a) \vdash \exists z.Q(z)}\, \exists\colon r}{P(a) \vee Q(b), \forall x.\neg P(x) \vdash \exists z.Q(z)}\, \forall\colon l$$

Here we see that z was replaced by b in $\exists\colon r$ (reading the proof backwards); moreover ψ contains a so-called *Herbrand sequent*

$$S_H\colon P(a) \vee Q(b), \neg P(a) \vdash Q(b)$$

which is valid and can be obtained by instantiation of the quantified formulas from the end-sequent. So we get an *explicit* information about the "right" z in the cut-free proof. Note that, in general, not a single witness is obtained but rather a set of witnesses (Herbrand "disjunction"). Gentzen [16] has given a proof transformation on cut-free proofs to obtain Herbrand sequents (which he called mid-sequents, because the upper part of the obtained proof consists only of structural and propositional, the lower only of quantifier- and structural rules). We describe this theorem in more detail in Section 3.

3 Proof-Theoretical Aspects of Cut-Elimination

The main and basic theorem of proof theory is the so-called *Hauptsatz*:

Theorem 1 (Gentzen 1934). *Let φ be an* **LK***-proof of a sequent S. Then there exists an* **LK***-proof ψ of S (effectively constructible from φ) without application of the cut-rule.*

Gentzen's proof is based on a proof transformation method which will be described in more detail below. A cut-free proof is a *normal form* under this transformations. We will illustrate the benefits of this normal form below.

Proof transformations to normal form (cut-free proofs in **LK**, normal proofs in natural deduction) essentially change the nature of the proof in making implicit information explicit (see the simple Example 1). In case of cut-free **LK**-proofs of prenex end-sequents we obtain a *Herbrand sequent* describing all instantiations of quantifiers in the proof and reducting the problem to a propositional one. This abstraction from propositional reasoning allows mathematical interpretations of complex cut-free proofs obtained via cut-elimination (see [20] and [31]). For simplicity we define Herbrand sequents for prenex sequents only. Instead of working with a sequent S we can consider the skolemized form $sk(S)$, a form where the so-called strong quantifiers are eliminated via the introduction of terms in a new signature. This transformation is standard in automated deduction and crucial to the transformation into clause form. Skolemization can also be applied to whole proofs. In fact, every proof of φ of S can be transformed in to a proof φ' of $sk(S)$ by a merely quadratic proof transformation (the skolemization of proofs, see [10]). On the other hand, every cut-free proof of $sk(S)$ can be transformed into a cut-free proof of S; this transformation is polynomial if S is prenex and exponential in general [5]. For simplicity we assume that the sequent S is in prenex form. Then $sk(S)$ is of the form $A_1, \ldots, A_n \vdash B_1, \ldots, B_m$ where the A_i are of the form $\forall x_1, \ldots, \forall x_k.E$ (a Π_1-formula), and the B_j of the form $\exists y_1, \ldots, \exists y_l.F$ (a Σ_1-formula), where E and F are quantifier-free . The specific form of these sequents motivates the following definition:

Definition 4. *A sequent $S\colon A_1, \ldots, A_n \vdash B_1, \ldots, B_m$, where the A_i are Π_1- and the B_j Σ_1-formulas is called a Σ_1-sequent.*

The essence of Herbrand's theorem consists in the replacement of quantified formulas by instances of these formula, resulting in a quantifier-free formula which is validity-equivalent. We formulate this theorem in form of prenex sequents.

Definition 5. *Let A be a Σ_1- or a Π_1-formula of the form $Qx_1, \ldots, Qx_n.E$ and $t_1, \ldots, t_n$ be terms. Then $E\{x_1/t_1, \ldots, x_n/t_n\}$ is called an instantiation of A.*

Definition 6 (Herbrand sequent). *Let $S\colon A_1, \ldots, A_n \vdash B_1, \ldots, B_m$ be a provable Σ_1-sequent. For any A_i (B_j) let $\mathcal{A}_i$ ($\mathcal{B}_j$) be a sequences of instantiations of $\mathcal{A}_i$ ($\mathcal{B}_j$). Then $S'\colon \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B}_1, \ldots, \mathcal{B}_m$ is called a* Herbrand sequent *of S if S' is propositionally valid.*

Herbrand sequents can be constructed from proofs. We first eliminate all cuts down to atomic ones (full cut-elimination is not required) and then construct a sequent consisting only of instances of formulas of the end-sequent.

Theorem 2 (mid-sequent theorem). *Let φ be an* **LK***-proof of a Σ_1-sequent S with at most atomic cuts (if φ contains cuts then they are atomic). Then φ*

can be transformed into a proof φ' with the same number of logical inferences and atomic cuts and with the following property. φ' contains the derivation of a sequent S' (the mid-sequent), s.t. all propositional inferences and atomic cuts in φ are above S', and below S' there are only unary structural rules and quantifier-rules from φ.

Proof. In [16] a step-wise proof transformation is given transforming a proof of a prenex sequent into a proof containing a mid-sequent. If we are only interested in the mid-sequent itself, which by our definition is a Herbrand sequent, it suffices to read off the instances from the quantifier-introduction rules and collect them in a sequent. This procedure can be performed in linear time.

Corollary 1. *Let φ be a proof of a Σ_1-sequent S with at most atomic cuts and let S' be a mid-sequent as defined in Theorem 2. Then S' is a Herbrand sequent of S.*

Proof. Let φ' be the proof obtained from φ by the transformation of Theorem 2. Then the subproof ψ of φ' deriving the midsequent S' contains only propositional and structural rules. By the soundness of **LK** S' is propositionally valid.

Example 2. Let ψ be the proof from Example 1:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(a) \vdash P(a)}{P(a), \neg P(a) \vdash}\,\neg\colon l}{P(a), \neg P(a) \vdash Q(b)}\,w\colon r \quad \dfrac{Q(b) \vdash Q(b)}{Q(b), \neg P(a) \vdash Q(b)}\,w\colon l}{P(a) \vee Q(b), \neg P(a) \vdash Q(b)}\,\vee\colon l}{P(a) \vee Q(b), \neg P(a) \vdash \exists z.Q(z)}\,\exists\colon r}{P(a) \vee Q(b), \forall x.\neg P(x) \vdash \exists z.Q(z)}\,\forall\colon l$$

This proof is already in midsequent form and does not need any transformation. Its midsequent is

$$S'\colon P(a) \vee Q(b), \neg P(a) \vdash Q(b).$$

S' is a Herbrand sequent of $P(a) \vee Q(b), \forall x.\neg P(x) \vdash \exists z.Q(z)$.

The concept of Herbrand sequent can be generalized to non-prenex sequents (see [6]); efficient algorithms for the computation of these more general sequents are given in [31]. Herbrand sequents can also be represented as expansion trees [24]. The construction of Herbrand sequents works not only for cut-free proofs but also with proofs containing only cuts with quantifier-free formulas. This is one of the reasons why quantifier-free cuts are frequently called "inessential" [29]. So the most essential elimination is this of *quantified* cuts. In the method `ceres` defined below, we transform arbitrary **LK**-proofs into **LK**-proofs with only atomic cuts; these cuts are inessential and we may speak about cut-elimination, even if inessential cuts are still present.

As cut-elimination is of high (in the worst-case nonelementary complexity) the specific choice of algorithms is crucial. The proof reduction method

of Gentzen (defined by reductions in the corresponding proof rewrite system, see [10]) turns out to be very redundant and expensive. The radically different method `ceres` (cut-elimination by resolution) has been developed in [7] and [9]. `ceres` is a cut-elimination method that is based on resolution. The method roughly works as follows: The structure of the proof containing cuts is mapped to a clause term which evaluates to an unsatisfiable set of clauses $\mathcal{C}$ (the *characteristic clause set*). A resolution refutation of $\mathcal{C}$, which is obtained using a first-order theorem prover, serves as a skeleton for the new proof which contains only atomic cuts. In a final step also these atomic cuts can be eliminated, provided the (atomic) axioms are valid sequents (or, at least, are closed under cut); but this step is of minor mathematical interest only. In the system CERES[4] this method of cut-elimination has been implemented. The system is capable of dealing with formal proofs in **LK**, among them also very large ones; moreover it was used in the analysis of Fürstenberg's proof of the infinitude of primes to be described in Section 5.

Gentzen's proof of cut-elimination in **LK** is based on a reduction relation which selects an uppermost cut and reduces its complexity. There are two possibilities:

1. the cut formulas on both sides are introduced by rules immediately over the cut in the proof. Then the cut is simplified to one or two cuts of lower formula complexity (or the cut is deleted at all). This is called a *grade* reduction.
2. One or both of the cut formulas are not introduced immediately above the cut. Then the cut is shifted upwards and a rule permutation is performed. This reduction is called a *rank* reduction. Rank reduction rules serve the purpose to enforce a situation in which a grade reduction rule can be carried out.

From the rank and grade reductions in Gentzen's proof a set of proof rewrite rules $\mathcal{R}$ can be extracted. Further refinements of $\mathcal{R}$, e.g. restricting the rewriting to uppermost cuts in the proof (Gentzen reduction $>_G$) guarantee that the rewriting relation is terminating. A terminating sequence of proof reduction yields a so-called Gentzen normal form (which is not unique as $>_G$ is not confluent). For the complete list of $\mathcal{R}$ we refer to [10]; here we list just two of them to illustrate the nature of the method:

– a rank-reduction rules which shifts the cut rule over an $\vee\colon l$-rule. The proof

$$\dfrac{\dfrac{\dfrac{(\varphi_1)}{B, \Gamma \vdash \Delta, A} \quad \dfrac{(\varphi_2)}{C, \Gamma \vdash \Delta, A}}{B \vee C, \Gamma \vdash \Delta, A} \vee\colon l \quad \dfrac{(\psi)}{A, \Pi \vdash \Lambda}}{B \vee C, \Gamma, \Pi \vdash \Delta, \Lambda}\ cut$$

[4] available at `http://www.logic.at/ceres/`

reduces to

$$\dfrac{\dfrac{\overset{(\varphi_1)}{B, \Gamma \vdash \Delta, A} \quad \overset{(\psi)}{A, \Pi \vdash \Lambda}}{B, \Gamma, \Pi \vdash \Delta, \Lambda}\ cut \quad \dfrac{\overset{(\varphi_2)}{C, \Gamma \vdash \Delta, A} \quad \overset{(\psi)}{A, \Pi \vdash \Lambda}}{C, \Gamma, \Pi \vdash \Delta, \Lambda}\ cut}{B \vee C, \Gamma, \Pi \vdash \Delta, \Lambda} \vee\colon l$$

– A grade reduction rules which reduces the logical complexity of the cut formula. The proof

$$\dfrac{\dfrac{\overset{(\varphi_1)}{\Gamma \vdash \Delta, A(x/t)}}{\Gamma \vdash \Delta, \exists x.A(x)} \exists\colon r \quad \dfrac{\overset{(\varphi_2(y))}{A(x/y), \Pi \vdash \Lambda}}{\exists x.A(x), \Pi \vdash \Lambda} \exists\colon l}{\Gamma, \Pi \vdash \Delta, \Lambda}\ cut$$

reduces to

$$\dfrac{\overset{(\varphi_1)}{\Gamma \vdash \Delta, A(x/t)} \quad \overset{(\varphi_2(t))}{A(x/t), \Pi \vdash \Lambda}}{\Gamma, \Pi \vdash \Delta, \Lambda}\ cut$$

Here, y is an eigenvariable which does not occur in Π, Λ, but occurs free in the proof $\varphi_2(y)$. $\varphi_2(t)$ is defined by replacing y in $\varphi_2(y)$ by t. To ensure soundness of this proof substitution t may not contain variables bound in φ_2; this property can be guaranteed by distinguishing free and bound variables [16].

Example 3. We consider the proof φ in Example 1 (where some subproofs are abbreviated by $\varphi_1, \varphi_2, \varphi_3$):

$$\dfrac{\dfrac{\dfrac{\overset{(\varphi_1)}{P(a) \vdash P(a) \vee Q(a)}}{P(a) \vdash \exists y(P(y) \vee Q(y))} \exists\colon r \quad \overset{(\varphi_2)}{Q(b) \vdash \exists y(P(y) \vee Q(y))}}{P(a) \vee Q(b) \vdash \exists y(P(y) \vee Q(y))} \vee\colon l \quad \dfrac{\overset{(\varphi_3(\alpha))}{P(\alpha) \vee Q(\alpha), \forall x.\neg P(x) \vdash \exists z.Q(z)}}{\exists y(P(y) \vee Q(y)), \forall x.\neg P(x) \vdash \exists z.Q(z)} \exists\colon l}{P(a) \vee Q(b), \forall x.\neg P(x) \vdash \exists z.Q(z)}\ cut$$

We see that the cut formula is immediately introduced in the right side of the proof, but not in the left one. Therefore we apply the rank reduction rule for $\vee\colon l$ defined above and obtain the proof $\varphi' =$

$$\dfrac{\dfrac{\dfrac{\overset{(\varphi_1)}{P(a) \vdash P(a) \vee Q(a)}}{P(a) \vdash \exists y(P(y) \vee Q(y))} \exists\colon r \quad \dfrac{\overset{(\varphi_3(\alpha))}{P(\alpha) \vee Q(\alpha), \forall x.\neg P(x) \vdash \exists z.Q(z)}}{\exists y(P(y) \vee Q(y)), \forall x.\neg P(x) \vdash \exists z.Q(z)} \exists\colon l}{P(a), \forall x.\neg P(x) \vdash \exists z.Q(z)}\ cut \quad \overset{(\eta)}{Q(b), \forall x.\neg P(x) \vdash \exists z.Q(z)}}{P(a) \vee Q(b), \forall x.\neg P(x) \vdash \exists z.Q(z)} \vee\colon l$$

where $\eta =$

$$\dfrac{\overset{(\varphi_2)}{Q(b) \vdash \exists y(P(y) \vee Q(y))} \quad \dfrac{\overset{(\varphi_3(\alpha))}{P(\alpha) \vee Q(\alpha), \forall x.\neg P(x) \vdash \exists z.Q(z)}}{\exists y(P(y) \vee Q(y)), \forall x.\neg P(x) \vdash \exists z.Q(z)} \exists\colon l}{Q(b), \forall x.\neg P(x) \vdash \exists z.Q(z)}\ cut$$

Now we locate the leftmost uppermost cut in φ'; the corresponding subproof ψ is

$$\dfrac{\dfrac{\dfrac{(\varphi_1)}{P(a) \vdash P(a) \vee Q(a)}}{P(a) \vdash \exists y(P(y) \vee Q(y))}\ \exists\colon r \quad \dfrac{\dfrac{(\varphi_3(\alpha))}{P(\alpha) \vee Q(\alpha), \forall x.\neg P(x) \vdash \exists z.Q(z)}}{\exists y(P(y) \vee Q(y)), \forall x.\neg P(x) \vdash \exists z.Q(z)}\ \exists\colon l}{P(a), \forall x.\neg P(x) \vdash \exists z.Q(z)}\ cut$$

In the next step we obtain the proof φ'' by replacing ψ by ψ', which is obtained via the grade reduction rule for $\exists$-cuts defined above. ψ' is

$$\dfrac{\dfrac{(\varphi_1)}{P(a) \vdash P(a) \vee Q(a)} \quad \dfrac{(\varphi_3(a))}{P(a) \vee Q(a), \forall x.\neg P(x) \vdash \exists z.Q(z)}}{P(a), \forall x.\neg P(x) \vdash \exists z.Q(z)}\ cut$$

So, in one part of the proof φ' we have broken down a cut with $\exists y(P(y) \vee Q(y)$ to a cut with $P(a) \vee Q(a)$, which is of lower logical complexity. We can do a similar thing for the remaining cut in φ'' with $\exists y(P(y) \vee Q(y)$ in φ'' (with a new cut formula $P(b) \vee Q(b)$). There are several steps more before all cuts are eliminated.

The method `ceres` is based on a totally different approach: We analyze the proof φ first and extract a structure from the binary rules in the proof, the characteristic clause set $\mathrm{CL}(\varphi)$. In the second step we compute the so-called proof projections to the clauses in $\mathrm{CL}(\varphi)$; these are cut-free proofs obtained by skipping rules inferring ancestors of the cut rule in φ. The third step consists in a resolution refutation of $\mathrm{CL}(\varphi)$. he last one in plugging the resolution refutation together with the projections; this yields a proof with only atomic cuts. Below we give a rather informal description of `ceres` (but we will provide a formal definition of the characteristic clause set, which is the most important structure within the `ceres`-method).

Example 4. Let us consider again the proof φ from Example 1:

$$\dfrac{\dfrac{\dfrac{\dfrac{P(a) \vdash P(a)^\star}{P(a) \vdash P(a) \vee Q(a)^\star}\ \vee\colon r_1}{P(a) \vdash \exists y(P(y) \vee Q(y))^\star}\ \exists\colon r \quad \dfrac{\dfrac{Q(b) \vdash Q(b)^\star}{Q(b) \vdash P(b) \vee Q(b)^\star}\ \vee\colon r_2}{Q(b) \vdash \exists y(P(y) \vee Q(y))^\star}\ \exists\colon r}{P(a) \vee Q(b) \vdash \exists y(P(y) \vee Q(y))^\star}\ \vee\colon l \quad \overset{(\chi)}{\exists y(P(y) \vee Q(y))^\star, \forall x.\neg P(x) \vdash \exists z.Q(z)}}{P(a) \vee Q(b), \forall x.\neg P(x) \vdash \exists z.Q(z)}\ cut$$

for $\chi =$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(\alpha)^\star \vdash P(\alpha)}{P(\alpha)^\star, \neg P(\alpha) \vdash}\ \neg\colon l}{P(\alpha)^\star, \neg P(\alpha) \vdash Q(\alpha)}\ w\colon r \quad \dfrac{Q(\alpha)^\star \vdash Q(\alpha)}{Q(\alpha)^\star, \neg P(\alpha) \vdash Q(\alpha)}\ w\colon l}{P(\alpha) \vee Q(\alpha)^\star, \neg P(\alpha) \vdash Q(\alpha)}\ \vee\colon l}{P(\alpha) \vee Q(\alpha)^\star, \neg P(\alpha) \vdash \exists z.Q(z)}\ \exists\colon r}{P(\alpha) \vee Q(\alpha)^\star, \forall x.\neg P(x) \vdash \exists z.Q(z)}\ \forall\colon l}{\exists y(P(y) \vee Q(y))^\star, \forall x.\neg P(x) \vdash \exists z.Q(z)}\ \exists\colon l$$

where all cut-ancestors were marked with $\star$. We trace the ancestors up to the axioms where we find $C_1\colon\ \vdash P(a), C_2\colon\ \vdash Q(b)$ on the left-hand-side, and $C_3\colon P(\alpha)\vdash, C_4\colon Q(\alpha)\vdash$ on the right-hand-side. There is one binary inference $\vee\colon l$ on the left side, which goes into the end-sequent, by which we merge C_1, C_2 to $C_5\colon\ \vdash P(a), Q(b)$. On the right side of the proof the binary inference $\vee\colon l$ operates on ancestors of the cut (and does go into the cut), and we union the clauses to the set $\{C_3\colon P(\alpha)\vdash, C_4\colon Q(\alpha)\vdash\}$. Finally the cut itself is binary rule "going into the cut", and we take the union of all clauses generated so far; the result is

$$\mathcal{C} = \{\vdash P(a), Q(b),\quad P(\alpha)\vdash,\quad Q(\alpha)\vdash\}.$$

Note that α is a variable. $\mathcal{C}$ is called the *characteristic clause set* of φ. $\mathcal{C}$ is unsatisfiable and has the following resolution refutation R:

$$\dfrac{Q(\beta)\vdash \qquad \dfrac{\vdash P(a), Q(b)\quad P(\alpha)\vdash}{\vdash Q(b)}}{\vdash}$$

Note that, as common in resolution theorem proving, we may always rename the variables in a clause. By applying the substitution $\Theta = \{\alpha \to a, \beta \to b\}$ we obtain a propositional refutation R' of $\{\vdash P(a), Q(b),\ P(a)\vdash,\ Q(b)\vdash\}$ of the form

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

This proof R' can be taken as a *skeleton* of a proof with only atomic cuts of the end sequent $S\colon P(a)\vee Q(b), \forall x.\neg P(x)\vdash \exists z.Q(z)$. In this skeleton will fill in the so-called proof projections; these are cut-free proofs of the end-sequent + an instance of a characteristic clause. The idea of a proof projection is to skip all inferences going into the cut; all inferences going into the end-sequent are performed. Skipping binary rules going into the cut is achieved by weakening. We consider the clause $C_5\colon\ \vdash P(a), Q(b)$ and the corresponding projection π_1 (built from the left part of the proof):

$$\dfrac{\dfrac{\dfrac{P(a)\vdash P(a)}{P(a)\vdash P(a), Q(b)}\ w\colon r \qquad \dfrac{Q(b)\vdash Q(b)}{Q(b)\vdash P(a), Q(b)}\ w\colon r}{P(a)\vee Q(b)\vdash P(a), Q(b)}\ \vee\colon l}{P(a)\vee Q(b), \forall x.\neg P(x)\vdash \exists z.Q(z), P(a), Q(b)}\ w^*$$

From the right part of the proof and the instances $P(a)\vdash$ and $Q(b)\vdash$ of the clauses C_3, C_4 we get (by instantiating α in φ by a and b) the projections $\pi_2 =$

$$\dfrac{\dfrac{\dfrac{P(a)\vdash P(a)}{P(a), \neg P(a)\vdash}\ \neg\colon l}{P(a), \forall x.\neg P(x)\vdash}\ \forall\colon l}{P(a), P(a)\vee Q(b), \forall x.\neg P(x)\vdash \exists z.Q(z)}\ w^*$$

and $\pi_3 =$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{Q(b) \vdash Q(b)}{Q(b), \neg P(a) \vdash Q(b)}\, w\colon l}{Q(b), \neg P(a) \vdash \exists z.Q(z)}\, \exists\colon r}{Q(b), \forall x.\neg P(x) \vdash \exists z.Q(z)}\, \forall\colon l}{Q(b), P(a) \vee Q(b), \forall x.\neg P(x) \vdash \exists z.Q(z)}\, w\colon l$$

Let $S \circ C$ be the sequent S merged with the sequent C. Then the projections are cut-free proofs π_1 of $S\circ \vdash P(a), P(b)$, π_2 of $P(a)\vdash \circ S$ and π_3 of $Q(b)\vdash \circ S$. By replacing the clauses in R' by the proofs π_1, π_2, π_3 we get the proof:

$$\dfrac{\dfrac{\dfrac{(\pi_3)}{Q(b) \vdash \circ S} \qquad \dfrac{\dfrac{(\pi_1)}{\vdash P(a), Q(b) \circ S} \qquad \dfrac{(\pi_2)}{P(a) \vdash \circ S}}{\vdash Q(b) \circ S \circ S}\, cut}{S \circ S \circ S}\, cut}{S}\, c^*$$

where c^* denotes a sequence of contractions.

Note that the propositional resolution and atomic cut are the same rules; indeed, in our formalism, propositional resolution is just a sub-calculus of **LK**.

The general definition of a characteristic clause set is the following:

Definition 7. *We define clause sets $\mathcal{C}_\nu$ for every node of a prof tree:*

- *in the axioms S select the subsequents S' consisting of atoms which are ancestors of a cut, and construct $\{S'\}$.*
- *Assume that a clause set $\mathcal{C}_\nu$ is already constructed at premise node ν of a unary inference yielding the conclusion ν'. Then $\mathcal{C}_{\nu'} = \mathcal{C}_\nu$ (unary inferences do not change the clause set).*
- *Assume that $\mathcal{C}_{\nu_1}$ and $\mathcal{C}_{\nu_2}$ for the premises ν_1, ν_2 are already defined. We distinguish two cases:*
  - *The inferred formula in the consequent ν is an ancestor of a cut (or does not exist as the rule itself is a cut). Then $\mathcal{C}_\nu = \mathcal{C}_{\nu_1} \cup \mathcal{C}_{\nu_2}$.*
  - *The inferred formula in the consequent ν is an ancestor of the end-sequent. Then $\mathcal{C}_\nu = \mathcal{C}_{\nu_1} \times \mathcal{C}_{\nu_2}$, where $\mathcal{C} \times \mathcal{D} = \{C \circ D \mid C \in \mathcal{C}, D \in \mathcal{D}\}$.*

If ν_0 is the root of the proof φ then $\mathrm{CL}(\varphi) = \mathcal{C}_{\nu_0}$. $\mathrm{CL}(\varphi)$ *is called the* characteristic clause set *of φ*

For a formal definition of the proof projections we refer to [9] and [10]. As already illustrated in Example 4 a projection of a proof φ of S to a characteristic clause C is a cut-free proof of the sequent $S \circ C$ with less inferences than φ. After the construction of a resolution refutation R (in form of a proof tree) of $\mathrm{CL}(\varphi)$, the global most general unifier ϑ of the resolutions is applied to R; the resulting propositional resolution tree R' is an **LK**-refutation tree of instances of the characteristic clause set. R' is turned into a proof of S with atomic cuts by replacing the clauses by the proof projections. The final proof with atomic cuts

is called a `ceres`-*normal form* of φ. `ceres`, in its original version, requires proof skolemization as a preprocessing; without skolemization of the proof projections may violate eigenvariable conditions. However, the extension of `ceres` to higher-order logic (see [19]) yields (quasi as a side effect) a version of first-order `ceres` without skolemization, which is relatively complex (locally unsound violations of eigenvariable conditions can be repaired globally by a proof transformation).

A comparison of `ceres` and the Gentzen method (and, more general, of every cut-elimination method using the Gentzen proof-rewriting rules $\mathcal{R}$) shows that `ceres` is capable of producing much shorter proofs in the following an exact asymptotic sense, that of nonelementary improvement. Nonelementary improvement (to be formally defined below) is a natural measure in comparing cut-elimination methods as the complexity of cut-elimination itself is nonelementary.

Definition 8. *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* $(l_1, \ldots, l_k)$ *is less than or equal to* $e(n, |(l_1, \ldots, l_k)|)$ *where* $|\ |$ *denotes the maximum norm on* $\mathbb{N}^k$ *(see also [13]).*

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.

Remark 1. The notion of elementary function is robust under use of different models of Turing machines. In fact, it does not matter whether we consider machines with just one tape or several ones, or machines with unary or k-ary alphabets for $k > 1$.

Note that the functions s and e are nonelementary. In general, any function f which grows "too fast", i.e. for which there exists no number k s.t.

$$f(n) \leq e(k, n),$$

is nonelementary.

Every exponential function $f(x, y)$ of the form $p(x)^{q(y)}$ for polynomials p and q is elementary. It is easy to prove that there exists a Turing machine T computing f and number k s.t. the computing time of T on (x, y) is less than $e(k, |(x, y)|)$.

Definition 9. *Let* $\zeta\colon (x_n)_{n\in\mathbb{N}}$ *and* $\eta\colon (y_n)_{n\in\mathbb{N}}$ *two sequences of natural numbers. We say that* ζ *is* elementary in η *if there exists a number* k *s.t. for all* $n \in \mathbb{N}$*:* $x_n \leq e(k, y_n)$*; otherwise* ζ *is called* nonelementary in η.

For complexity analysis we use two measures:

– the symbolic complexity $\| \ \|$ (the number of symbol occurrences), and
– $l(\psi)$, the length of proof ψ (the number of inference nodes in the proof tree)

In [28] R. Statman proved that there exists a sequence of short proofs γ_n of sequents $S_n\colon \Gamma \vdash A_n$ s.t. the Herbrand complexity $\mathrm{HC}(S_n)$ of S_n (which is the minimal symbol complexity of a Herbrand sequent of a cut-free proof of S_n) is inherently nonelementary in $l(\gamma_n)$ (it is also nonelementary in $\|\gamma_n\|$); in fact Statman did not explicitly address a specific formal calculus, leaving the formalization of the proof sequence to the reader. Independently V. Orevkov [25] proved the nonelementary complexity of cut-elimination for function-free predicate logic without equality. The proof sequences of Statman and Orevkov are different, but both encode the principle of iterated exponentiation best described by P. Pudlak in [26].

Theorem 3 (Statman, Orevkov). *There exists a sequence S_n of sequents with the following properties:*

– *There is a constant a s.t. for every n there exists an* **LK***-proof φ_n of S_n with $\|\varphi_n\| \leq 2^{a*n}$.*
– *For every n let $c(n) = \min\{\|\psi\| \mid \psi$ is a cut-free proof of $S_n\}$. Then $(c_n)_{n\in\mathbb{N}}$ is not elementary in $\|\varphi_n\|$.*

Proof. In [28] and [25]. Note that the Herbrand complexity $\mathrm{HC}(S_n)$ of S_n defines a lower bound on $c(n)$; on the other hand, $c(n)$ is at most exponential (and thus elementary) in $\mathrm{HC}(S_n)$). So it does not matter, whether we speak of the symbolic lengths of shortest cut-free proofs or about Herbrand complexity.

Below we give a definition which gives a basis for comparing reductive cut-elimination methods and `ceres`. Thereby, reductive cut-elimination is described as sequence of proofs θ obtained via a proof reduction relation $>_x$ based on $\mathcal{R}$, starting with a proof φ and ending in a proof φ' with at most atomic cuts. Such a sequence θ is called an $>_x$-cut-elimination sequence on φ.

Definition 10. *Let $>_x$ be a proof reduction relation based on $\mathcal{R}$. We say that* `ceres` *NE-improves $>_x$ if there exists a sequence of proofs $(\varphi_n)_{n\in\mathbb{N}}$ s.t.*

– *there exists a sequence of resolution refutations $(\gamma_n)_{n\in\mathbb{N}}$ of the sequence of the corresponding characteristic clause sets $(\mathrm{CL}(\varphi_n))_{n\in\mathbb{N}}$ such that $(\|\gamma_n\|)_{n\in\mathbb{N}}$ is elementary in $(\|\varphi_n\|)_{n\in\mathbb{N}}$.*
– *For every n let $g(n) = \min\{\|\theta\| \mid \theta$ is an $>_x$ $-$cut-elimination sequence on $\varphi_n\}$. Then $(g(n))_{n\in\mathbb{N}}$ is nonelementary in $\|\varphi_n\|$.*

Similarly we define that $>_x$ NE-improves `ceres` *if there exists a sequence of proofs $(\varphi_n)_{n\in\mathbb{N}}$ s.t.*

– *there exists a sequence of $>_x$-cut-elimination sequences $(\theta_n)_{n\in\mathbb{N}}$ on $(\varphi_n)_{n\in\mathbb{N}}$ s.t. $(\|\theta_n\|)_{n\in\mathbb{N}}$ is elementary in $(\|\varphi_n\|)_{n\in\mathbb{N}}$.*
– *For all n let $h(n) = \min\{\|\gamma\| \mid \gamma$ is a resolution refutation of $\mathrm{CL}(\varphi_n)\}$. Then $(h(n))_{n\in\mathbb{N}}$ is nonelementary in $(\|\varphi_n\|)_{n\in\mathbb{N}}$.*

Remark 2. Comparing the size of the resolution refutations in `ceres` with the total size of cut-elimination sequences is justified, as the resolution refutations of characteristic clause sets are the main source of complexity in `ceres`; in fact, the computation time of a sequence of `ceres` normal forms grows nonelementarily in the size of the input proofs iff this holds for the computation of the resolution refutations. So, for this asymptotic comparison, the computation of the characteristic clause sets and the projections do not matter. Also mathematically the core of the `ceres`-method is the resolution refutation of the characteristic clause set.

Theorem 4. `ceres` *NE-improves the Gentzen method of cut-elimination*

Proof. We give a modified version of the proof in [10]. Let $(\psi_n)_{n\in\mathbb{N}}$ be a sequence of proofs for $\psi_n =$

$$\dfrac{\dfrac{\dfrac{A \vdash A}{A, \Delta_n \vdash A}\, w\colon l \quad \dfrac{\overset{(\gamma_n)}{\Delta_n \vdash D_n}}{A, \Delta_n \vdash D_n}\, w\colon l}{A, \Delta_n \vdash A \wedge D_n}\, \wedge\colon r \quad \dfrac{\dfrac{A \vdash A \quad A \vdash A}{A, A \to A \vdash A}\, \to\colon l}{A \wedge D_n, A \to A \vdash A}\, \wedge\colon l}{A, \Delta_n, A \to A \vdash A}\, \mathit{cut}$$

where γ_n is Statman's worst-case sequence admitting only nonelementary cut-elimination (no matter which method is applied); for details in the definition of γ_n see [10]. In the method of Gentzen we always select an uppermost cut. As all cuts in γ_n are above the cut with $A \wedge D_n$, Gentzen's method eliminates all the cuts in γ_n before eliminating the cut with formula $A \wedge D_n$; thus it constructs a cut-free proof of $\Delta_n \vdash D_n$, which is of nonelementary size in $\|\gamma_n\|$ and also in $\|\psi_n\|$.

Let us turn to `ceres` on ψ_n. The characteristic clause sets are

$$\mathrm{CL}(\psi_n) = \{\vdash A;\ A \vdash\} \cup \mathrm{CL}(\gamma_n).$$

Trivially every $\mathrm{CL}(\psi_n)$ has the resolution refutation $\rho =$

$$\dfrac{\vdash A \quad A \vdash}{\vdash}$$

which is of constant length and, by defining $\rho_n = \rho$ for all n, we get $\|\rho_n\| = 5$. Trivially $(\|\rho_n\|)_{n\in\mathbb{N}}$ is elementary in $(\|\psi_n\|)_{n\in\mathbb{N}}$.

A similar result also holds for the Tait-method, another method of cut-elimination based on $\mathcal{R}$ [10]. A nonelementary speed-up in the other direction is impossible – for every method based on $\mathcal{R}$.

Theorem 5. *No reductive method based on $\mathcal{R}$ NE-improves* `ceres`*; in particular the Gentzen method does not NE-improve* `ceres`*.*

Proof. In [9] and [10].

The proof of this theorem is based on a result showing that reductive cut-elimination has only redundant effects on the characteristic clause set. Surprisingly, this redundancy is defined by subsumption, a common redundancy-elimination principle of automated deduction (see e.g. [22]):

Theorem 6. *Let φ be an* **LK***-derivation and ψ be a normal form of φ under a cut reduction relation $>_{\mathcal{R}}$ based on $\mathcal{R}$. Then* $\mathrm{CL}(\varphi) \leq_{ss} \mathrm{CL}(\psi)$.

Proof. In [9] and [10].

The theorems above show that, from a complexity theoretic point of view, `ceres` is superior to reductive methods of cut-elimination. It pays out that, prior to cut-elimination, we analyze the proof and make use of the extracted structure of characteristic clause set (which then is analyzed via a resolution refutation). In contrast, the reductive methods are just local (they focus on the upmost operators of cut-formulas) and cannot take into account the global structure of proofs. `ceres` is much less redundant than the reductive methods, but also "less" confluent (note than also Gentzen's method is not confluent). In fact, `ceres` can produce much more different normal forms (and corresponding Herbrand sequents) than reductive methods. Section 4 will illustrate that this non-confluence can lead to interesting mathematical arguments extractable from proofs.

`ceres` is basically a method for cut-elimination in *classical logic*. The generalization to finitely valued logics is unproblematic [8]; there exists also a `ceres`-method for Gödel logic [2] and for subclasses of intuitionistic logic [23]. An advantage of reductive methods is their flexibility concerning structural restrictions; note that the reductive Gentzen method is virtually the same for classical and for intuitionistic logic.

4 Cut-Elimination in Practice

Using cut-elimination in practice requires first to *formalize* a mathematical proof as an **LK**-proof. In the formalization it is crucial to avoid unnecessary cuts, as then the characteristic clause sets becomes too complex, which can make it unfeasible for the theorem prover to refute them. This step is generally delicate as there is nothing like a unique formal representation of an informal mathematical proof. We address this problem once more in the application chapter where we describe the analysis of Fürstenberg's proof on the infinitude of primes. Reductive methods fail in practice because of the sheer size of the cut-free proofs and the high number of reduction steps. The `ceres` method, which is asymptotically superior as described in Section 3, is also much better in practice. This can be immediately seen by investigating the characteristic clause set of a real probelm. A typical characteristic clause set of an **LK**-proof with cuts contains numerous tautologies and subsumed clauses, which remain undetected by reductive methods. By `ceres`, however, which is based on resolution theorem proving, tautologies and subsumed clauses can be eliminated without loss of completeness. In fact,

the proofs obtained by Gentzen's method correspond to very redundant resolution derivations using lots of tautologies and subsumed clauses. Though the proofs found by `ceres` are generally much smaller and more compact, the size of the `ceres` normal forms still remains a barrier: remember that our aim is to analyze proofs, the main goal being the *interpretation* of the obtained cut-free proof. For huge `ceres` normal forms such an interpretation cannot be found by just reading down the proof. Further compressions of information are required. The ideal concept for compression is the Herbrand sequent of a proof constructible from all proofs with only atomic cuts (see 1). Herbrand sequents abstract from propositional reasoning and represent the first-order content of a proof in a very compact way; the Herbrand sequent of a proof essentially describes the instantiations of the formulas needed to prove the theorem. After this post-processing proofs become much more readable and it is much easier to mine the very mathematical content of it. In some cases (see the application chapter) the refutation of the characteristic clause alone contains the main mathematical information, and it is not even necessary to produce an atomic cut normal form or even a cut-free proof.

Gentzen's **LK** is the original calculus for which cut-elimination was defined. In formalizing mathematical proofs it turns out that **LK** (and also natural deduction) are not sufficiently close to real mathematical inference.

First of all, the calculus **LK** lacks a specific handling of equality to implement equality reasoning equality axioms have to be added to the end-sequent. Due to the importance of equality this defect was already apparent to proof theorists; e.g. Takeuti [29] defined an extension of **LK** to a calculus $\mathbf{LK}_=$, adding atomic equality axioms to the standard axioms of the form $A \vdash A$. The advantage of $\mathbf{LK}_=$ over **LK** is that no new axioms have to be added to the end-sequent; on the other hand, in presence of the equality axioms, full cut-elimination is no longer possible, but merely reduction to *atomic cut*. As we are not interested to eliminate atomic cuts this causes no problems. But still $\mathbf{LK}_=$ uses the same rules as **LK**; in fact, in $\mathbf{LK}_=$, equality is *axiomatized*, i.e. additional atomic (non-tautological) sequents are admitted as axioms. On the other hand, in formalizing mathematical proofs, using equality as a *rule* is much more natural and concise. For this reason we choose the most natural equality rule, which is strongly related to paramodulation in automated theorem proving. Our approach differs from this in [30], where a unary equality rule is used (which does not directly correspond to paramodulation). In the *equality rules* below we mark the auxiliary formulas by $+$ and the principal formula by $*$.

$$\frac{\Gamma_1 \vdash \Delta_1, s = t^+ \quad A[s]^+_\Lambda, \Gamma_2 \vdash \Delta_2}{A[t]^*_\Lambda, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2} =: l1 \quad \frac{\Gamma_1 \vdash \Delta_1, t = s^+ \quad A[s]^+_\Lambda, \Gamma_2 \vdash \Delta_2}{A[t]^*_\Lambda, \Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2} =: l2$$

for inference on the left and

$$\frac{\Gamma_1 \vdash \Delta_1, s = t^+ \quad \Gamma_2 \vdash \Delta_2, A[s]^+_\Lambda}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, A[t]^*_\Lambda} =: r1 \quad \frac{\Gamma_1 \vdash \Delta_1, t = s^+ \quad \Gamma_2 \vdash \Delta_2, A[s]^+_\Lambda}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, A[t]^*_\Lambda} =: r2$$

on the right, where Λ denotes a set of positions of subterms where replacement of s by t has to be performed. We call $s = t$ the *active equation* of the rules.

Furthermore, as the only axiomatic extension, we need the set of reflexivity axioms

$$\text{REF}: \quad \vdash s = s$$

for all terms s.

Definition 11. *The calculus* **LKe** *is* **LK** *extended by the axioms* REF *and by the rules*

$$=: l1, \ =: l2, \ =: r1 \ =: r2.$$

The calculus **LKe** contains additional rules and any cut-elimination method has to be adapted accordingly. For `ceres` this adaption consists in the extension of the clausal calculus from resolution to resolution + paramodulation. Note that **LKe**, without use of logical rules, coincides just resolution and paramodulation on clause logic – provided most general unification has already been carried out. The definitions of characteristic clause set and proof-projections remain the same, only we have to handle the equality rules as binary rules going into the end-sequent; for details see [10]. In contrast, the reduction rules of Gentzen cannot be adapted in an easy way. If we want to use reductive cut-elimination methods we must transform the whole proof prior to cut-elimination: all equality rules have to be shifted upwards in the proof s.t. they apply to atoms only. After this transformation the Gentzen method can be applied to the part of the proof below the equality rules to get a proof with only atomic cuts.

Also Herbrand sequent extraction can be generalized to **LKe**-proofs. We obtain a more general version of the midsequent theorem:

Theorem 7 (mid-sequent theorem for LKe). *Let φ be an* **LKe***-proof of a Σ_1-sequent S with at most atomic cuts s.t. all equality rules in φ are only applied to atoms. Then φ can be transformed into a proof φ' with the same number of logical inferences, equality rules and atomic cuts and with the following property: φ' contains the derivation of a sequent S' (the mid-sequent), s.t. all propositional inferences, atomic cuts and equational rules in φ are above S', and below S' there are only unary structural rules and quantifier-rules from φ.*

Proof. The proof transformation is essentially the same as in Theorem 2 as (still) the formulas in the end-sequent are prenex and there are no equality rules applied to quantified formulas.

Remark 3. Note that the sequents S' obtained in Theorem 7 are no longer valid in general, but just E-valid, i.e. valid in equality-interpretations (the predicate symbol $=$ is interpreted as equality over a domain). In fact, the equality rules of **LKe** are only valid w.r.t. equality-interpretations.

A further generalization is useful in practice. Instead of just axioms of the form $A \vdash A$ we may also allow equational axioms of the form $\vdash s = t$. Then the set of axioms is still consistent and its models are equational theories. The method `ceres` remains exactly the same as it can be applied to any proof with only atomic axioms. Moreover, for facilitating the specification of proofs,

definition-rules can be added to the calculus; the **LK**-version using the equality rules defined above and the definition rules is called **LKDe** (for details see [3]). Characteristic clause sets and their refutations are not affected by the extension by definitions.

The equational `ceres`-method based on **LKDe** has been applied to analyze several (rather simple) mathematical proofs fully automatic. We mention the analysis of the tape proof [3] and of the lattice proof [20]; in the latter one a Herbrand sequent was extracted and analyzed. For the tape proof different `ceres`-normal forms could be obtained (based on different refutations of the characteristic clause sets); their mathematical interpretations lead to different proofs in the sense, that the mathematical arguments were different (not just the form of the normal forms). In case of the lattice proof an equational Herbrand sequent was extracted, which clearly illustrated the mathematical argument behind the `ceres`-normal form. These applications illustrate that `ceres` is a suitable tool for *mining* proofs, i.e. to extract "hidden" mathematical information from proofs.

5 An Application of Cut-Elimination

We apply `ceres` to Fürstenberg's proof of the existence of infinitely many primes. The arguments of this proof are of topological nature, which form the synthetic notions of this synthetic proof. A natural formalization of this argument in second-order arithmetic is constructed and then translated to many-sorted first-order logic. In order to avoid induction axioms, the proof is eventually formalized as a scheme representing an infinite sequence of ordinary first-order proofs, demonstrating the existence of more and more primes. We show that the analytic proof schema corresponding to Euclid's proof belongs to the solution space of the schema of topological proofs.

In 1955 the renowned mathematician H. Fürstenberg published a proof of the infinity of primes by topological means [15] (see also [1]): He proved the infinity of primes using a topology induced by arithmetic progressions over the integers.

We give a proof with a topology over the natural numbers in order to have a simpler formulation of the proof later on. We start with the definition of a topological space:

Definition 12 (Topological Space). *A* topological space *is a set X together with a collection T of subsets of X satisfying the following axioms:*

1. *The empty set and X are in T.*
2. *The union of any collection of sets in T is also in T.*
3. *The intersection of any pair of sets in T is also in T.*

The collection T is called a topology *on X. The sets in T are the* open sets*, and their complements in X are the* closed sets*.*

The arithmetic progressions can be used as a basis for a topology over the natural numbers. We will denote an arithmetic progression by

$$\nu(a, b) = \{a + bn \mid n \in \mathbb{N}\}$$

for $a \in \mathbb{N}$ and $b \in \mathbb{N} \setminus \{0\}$.

Proposition 1. *By defining a set $A \subseteq \mathbb{N}$ as open, when A is either empty or for each $x \in A$ exists an $a \in \mathbb{N} \setminus \{0\}$ such that $\nu(x, a) \subseteq A$, one obtains a topology over $\mathbb{N}$.*

Proof. We check definition 12:

1. The empty set and $\mathbb{N}$ are open. Trivial.
2. The union of a collection of open sets is also open. Trivial.
3. The intersection of two open sets is also open.
   Let A and B two open sets. If $x \in A \cap B$, then there exist $a, b > 0$ such that $\nu(x, a) \subseteq A$ and $\nu(x, b) \subseteq B$ holds. Let c be the least common multiple of a and b, then $\nu(x, c) \subseteq \nu(x, a)$ and $\nu(x, c) \subseteq \nu(x, b)$, and hence $\nu(x, c) \subseteq A \cap B$.

A nice property of this topology is that every arithmetic progression starting at 0 is not only open but closed as well. Indeed this holds for every progression $\nu(a, b)$ where $a < b$, but this is not needed for the theorem.

Lemma 1. *Every arithmetic progression starting at 0 is closed.*

Proof. Let be $A = \nu(0, b)$ an arithmetic progression. Then the complement of A is a union of arithmetic progressions:

$$\bar{A} = \bigcup_{i=1}^{b-1} \nu(i, b).$$

The sets $\nu(i, b)$ are open, and the union of any collection of open sets is open; therefore $\bar{A}$ is open, hence A is closed.

Theorem 8. *There are infinitely many primes.*

Proof. Denote with P the set of all primes and assume P is finite. Let $X = \bigcup\{\nu(0, p) \mid p \in P\}$. By Lemma 1 every $\nu(0, p)$ for $p \in P$ is closed, so X is a finite union of closed sets and therefore closed as well. As every number different from 1 has a prime divisor we get $\bar{X} = \{1\}$. Being a complement of a closed set, $\bar{X}$ is open. But $\{1\}$ is neither empty nor does it contain an arithmetic progression, and so $\{1\}$ is not open. Contradiction! We conclude that P must be infinite.

The automated processing of Fürstenberg's proof requires a nontrivial logical preprocessing by humans. The first important step consists in the right choice of the logical language. As Fürstenberg's proof contains a topology defined over natural numbers and topological lemmas (with quantification over set-variables), an adequate candidate is *second-order arithmetic*. The formalization in [4] started

with a formalization of the proof in second-order arithmetic. In a second step this specification was translated into a scheme of sorted first-order definitions and proofs. For the details we refer to [4], but we present the main steps and formal definitions here. We start with the formalization in second-order arithmetic:

(a) $m \in \nu(k,l) \;\equiv\; \exists n(m = k + n * l)$.
(b) $\mathrm{DIV}(l,k) \equiv \exists m.l * m = k$.
(c) $\mathrm{PRIME}(k) \equiv 1 < k \wedge \forall l(\mathrm{DIV}(l,k) \to (l = 1 \vee l = k))$.
(d) $X \subseteq Y \equiv \forall n(n \in X \to n \in Y)$, and $X = Y \equiv X \subseteq Y \wedge Y \subseteq X$.
(e) $n \in \overline{X} \equiv n \notin X$.
(f) A function $p\colon \mathbb{N} \to \mathbb{N}$ which enumerates primes is one that fulfills the property:

$$\forall i \forall k(p(i) = k \to \mathrm{PRIME}(k)).$$

For the definition of p the comprehension principle is needed; for information about function definitions in second-order arithmetic see [27].
(g) $n \in \mathrm{S}[l] \equiv \exists m(m \leq l \wedge n \in \nu(0, p(m)))$.
$\mathrm{S}[l]$ describes the set of all elements n which occur in some $\nu(0,k)$, where k is one of the first $l+1$ primes enumerated by p. In mathematical notation we get

$$\mathrm{S}[l] = \bigcup_{m=0}^{l} \nu(0, p(m)).$$

(h) $\mathrm{F}[l] \equiv \forall k(\mathrm{PRIME}(k) \leftrightarrow \exists m(m \leq l \wedge k = p(m)))$.
$\mathrm{F}[l]$ is a formula which asserts that there are only $l+1$ primes, namely $\{p(0), \ldots, p(l)\}$.
(i) $\mathrm{O}(X) \equiv \forall m(m \in X \to \exists l\ \nu(m, l+1) \subseteq X)$.
(j) $\mathrm{C}(X) \equiv \mathrm{O}(\overline{X})$.
(k) $\infty(X) \equiv \forall k \exists l\ k + l + 1 \in X$.

Let (*) be the assumption that all primes occur in the set $M\colon\ \{p(0), \ldots, p(l)\}$. The first lemma in Fürstenberg's proof states that, under the assumption (*), every natural number different from 1 occurs in some $\nu(0,m)$ for $m \in M$. The corresponding formula is

$$\text{(I)}\ \forall l(\mathrm{F}[l]\ \to\ \mathrm{S}[l] = \overline{\{1\}}).$$

The second lemma states that, under the assumption (*), the set $\mathrm{S}[l]$ is closed. The formula expressing this lemma is

$$\text{(II)}\ \forall l(\mathrm{F}[l] \to \mathrm{C}(\mathrm{S}[l])).$$

Proofs of (I) and (II) can easily combined to a proof of

$$\text{(III)}\ \forall l(\mathrm{F}[l] \to \mathrm{C}(\overline{\{1\}})).$$

The proofs of (II) and (III) in second order arithmetic require induction. By (j) it is straightforward to prove

$$\text{(IV)}\ \forall l(\mathrm{F}[l] \to \mathrm{O}(\{1\})).$$

Another main lemma of the proof states that nonempty open sets are infinite:

$$\text{(V) } \forall X(\mathrm{O}(X) \wedge X \neq \emptyset \rightarrow \infty(X)).$$

While (I), (II), (III) and (IV) can be directly translated to first order logic (via the definitions), (V) is genuinely second order. Using (V) we show that $\infty(\{1\})$ holds giving a contradiction to $\neg\infty(\{1\})$, which is easily derivable in second order arithmetic.

To formulate Fürstenberg's proof in **LKDe** it is necessary to schematize it in order to avoid induction. In particular, induction is needed to prove the lemmas (II) and (III) above. The tool `hlk` [21] allows to define an infinite sequence of **LKDe**-proofs by specifying a proof scheme. The k-th proof can then be generated automatically from the scheme for any k.

The k-th proof shows that there cannot be $\leq k+1$ prime numbers.

To compile the second-order formulation to first order we work in a two-sorted logic containing sorts for 1. the natural numbers (denoted by $k, l, m, n, \ldots$ as before) and 2. sets of natural numbers (denoted by $x, y, \ldots$). Addition ($+$), multiplication ($*$) and the less-than relation ($<$) in the natural numbers are axiomatized. The background theory is purely universal and thus can be expressed as a set of clauses AX; It contains 34 clauses, among them associativity, commutativity and distributivity laws plus some derived laws like e.g. the cancellation law ($k + l = m + l \vdash k = m$). For the full list of axioms see the documentation on the web[5]. All of these axiom clauses are valid axiom sequents for the **LKDe**-proof.

Some of the definitions (a) to (k) given above can be taken over without change. This holds for (a), (b) and (c). For the others we get:

(d') $x \subseteq y \equiv \forall n(n \in x \rightarrow n \in y)$, and $x = y \equiv x \subseteq y \wedge y \subseteq x$. Here we only replaced the set variables by variables of the sort "set of natural numbers".
(e') $n \in \overline{x} \equiv n \notin x$.
(f') Instead of p we introduce a finite set $\mathrm{P}[k]$ defined by

$$\mathrm{P}[k] \equiv \{p_0\} \cup \cdots \cup \{p_k\}.$$

where the p_i are constant symbols denoting primes. Note that the k appearing in the definition is a metavariable, not an object variable as l in the definition of $\mathrm{F}[l]$ and $\mathrm{S}[l]$.
(g') $\mathrm{S}[k] \equiv \nu(0, p_0) \cup \cdots \cup \nu(0, p_k)$. Note that, in place of the object variable l in the definition (g), we have the metavariable k of the scheme.
(h') $\mathrm{F}[k] \equiv \forall m(\mathrm{PRIME}(m) \leftrightarrow m \in \mathrm{P}[k])$.
(i') $\mathrm{O}(x) \equiv \forall m(m \in x \rightarrow \exists l\ \nu(m, l+1) \subseteq x)$.
(j') $\mathrm{C}(x) \equiv \mathrm{O}(\overline{x})$.
(k') $\infty(x) \equiv \forall k \exists l\ k + l + 1 \in x$.

In order to avoid induction we also introduce three axioms (which can be proven in Peano arithmetic): (1) Every number greater than 0 has a predecessor,

[5] `http://www.logic.at/ceres/examples/prime.php`

(2) every number is in a remainder class modulo l and (3) every number has a prime divisor. These axioms will be carried down to the antecedent of the end sequent of the **LKDe**-proof.

(1) $\mathrm{PRE} \equiv \forall k(0 < k \to \exists m\ k = m + 1)$
(2) $\mathrm{REM} \equiv \forall l(0 < l \to \forall m \exists k(k < l \wedge m \in \nu(k, l)))$
(3) $\text{PRIME-DIV} \equiv \forall m(m \neq 1 \to \exists l(\mathrm{PRIME}(l) \wedge \mathrm{DIV}(l, m)))$

We now formulate a proof $\varphi_1(k)$ which proves the translation of (IV) above:

$\varphi_1(k) :=$

$$\dfrac{\dfrac{\dfrac{\vdots\ \psi_1(k)}{\mathrm{F}[k], \text{PRIME-DIV} \vdash \mathrm{S}[k] = \overline{\{1\}}} \quad \dfrac{\vdots\ \psi_2(k)}{\mathrm{F}[k], \mathrm{PRE}, \mathrm{REM} \vdash \mathrm{C}(\mathrm{S}[k])}}{\mathrm{F}[k], \Gamma \vdash \mathrm{C}(\overline{\{1\}})}\ =: r \quad \dfrac{\vdots}{\mathrm{C}(\overline{\{1\}}) \vdash \mathrm{O}(\{1\})}}{\mathrm{F}[k], \Gamma \vdash \mathrm{O}(\{1\})}\ cut$$

The proof $\psi_1(k)$ shows that if there are $\leq k + 1$ primes, then by the prime divisor axiom PRIME-DIV, the complement of all multiples of these primes is $\{1\}$, and the proof $\psi_2(k)$ demonstrates (under the assumption of $\leq k+1$ primes and the remainder axiom REM) that the set of these multiples is closed. With the help of these lemmas we can show that the set $\{1\}$ is open — if there are $\leq k + 1$ primes.

The proof φ_2 (which does not depend on k) shows that every (non-empty) open set is infinite. This lemma yields that, under the assumption of the set of primes being finite, the set $\{1\}$ must be either empty or infinite; of course neither is the case, which is easily shown. Hence we get our end-sequent, stating that there cannot be $\leq k + 1$ primes:

$\varphi(k) :=$

$$\dfrac{\dfrac{\dfrac{\vdots}{\vdash \{1\} \neq \emptyset} \quad \dfrac{\dfrac{\vdots\ \varphi_1(k)}{\mathrm{F}[k], \Gamma \vdash \mathrm{O}(\{1\})} \quad \dfrac{\dfrac{\vdots\ \varphi_2}{\vdash \forall x((\mathrm{O}(x) \wedge x \neq \emptyset) \to \infty(x))} \quad \vdots}{\mathrm{O}(\{1\}), \{1\} \neq \emptyset \vdash \infty(\{1\})}\ cut}{\{1\} \neq \emptyset, \mathrm{F}[k], \Gamma \vdash \infty(\{1\})}\ cut}{\mathrm{F}[k], \Gamma \vdash \infty(\{1\})}\ cut \quad \dfrac{\vdots}{\infty(\{1\}) \vdash}}{\mathrm{F}[k], \Gamma \vdash}\ cut$$

$$\dfrac{\mathrm{F}[k], \Gamma \vdash}{\underbrace{\text{PRIME-DIV}, \mathrm{PRE}, \mathrm{REM}}_{\Gamma} \vdash \neg \mathrm{F}[k]}\ \neg : r$$

The proof-schema φ_k above was then subjected to skolemization and the characteristic clause sets $\mathrm{CL}(\varphi_k)$ were computed for a large interval $[0, k]$ (for $k > 10$). The clause sets were surprisingly simple and could be strongly reduced in size by subsumption and tautology-deletion. The next (human-based) step

consisted in the generalization of the clause pattern. The resulting sequence of clause sets (after redundancy-elimination) was

$$\mathrm{CL}_r := \mathcal{C}_r \cup \mathrm{AX}$$

where

$$\mathcal{C}_r := A \cup \bigcup_{i=0}^{r} B_i \cup \{C_r\}$$

for

$$C_r := \quad \vdash m_0 = 1, s_1(m_0) = p_0, \ldots, s_1(m_0) = p_r,$$

$B_i :=$

$$\begin{array}{l}
0 < p_i \vdash p_i = s_7(p_i) + 1 \\
0 < p_i \vdash t_0 = s_5(p_i, t_0) + (s_6(p_i, t_0) * p_i) \\
0 < p_i, s_5(p_i, t_0) = 0 \vdash t_0 = 0 + (s_6(p_i, t_0) * p_i) \\
0 < p_i \vdash s_5(p_i, t_0) < p_i \\
t_0 = p_i, m_0 * n_0 = t_0 \vdash m_0 = 1, m_0 = t_0 \\
t_0 = p_i \vdash 1 < t_0 \\
t_0 = p_i, 1 = n_0 * t_0 \vdash
\end{array}$$

and $A :=$

$$\begin{array}{l}
\vdash m_0 = 1, s_1(m_0) * s_4(m_0) = m_0 \\
\vdash m_0 + (((k * (l_0 + (1 + 1))) + (l_0 * (m_0 + 1))) + 1) = \\
\qquad k + ((k + (m_0 + 1)) * (l_0 + 1)) \\
m_0 = k_0 + (r_0 * ((t_0 + 1) * (t_1 + 1))) \\
\qquad \vdash m_0 = k_0 + ((r_0 * (t_0 + 1)) * (t_1 + 1)) \\
m_0 = k_0 + (r_0 * ((t_0 + 1) * (t_1 + 1))) \\
\qquad \vdash m_0 = k_0 + ((r_0 * (t_1 + 1)) * (t_0 + 1)) \\
\vdash (((t_0 + 1) * t_1) + t_0) + 1 = (t_0 + 1) * (t_1 + 1)
\end{array}$$

For this structurally simple sequence of characteristic clause sets a schema of resolution refutations was defined. In this refutation schema the crucial (schematic) clause

$$E_r : 1 < t_r \vdash$$

for $t_r = p_0 * \ldots * p_r + 1$ was derivable.

By several steps of paramodulations E_r was transformed into the form $E'_r : 1 < (s_r + 1) + 1 \vdash$ for some term s_r. From axiom clauses one could derive the clause $G : \vdash 1 < (w+1)+1$ (w being a variable). G and E'_r finally resolve to $\vdash$, the empty clause. The term t_r obtained in the derivation by resolution and paramodulation

reflects exactly the construction in Euclid's proof of the infinitude of primes. We see that the elimination of topological arguments (performed by resolution and paramodulation in `ceres`) reveals the "true" character of Fürstenberg's proof, in the sense that it yield a *construction method for primes*, while the original proof does not.

6 Open Problems

As most mathematical proofs about numbers or discrete structures use inductive arguments, it is desirable to apply cut-elimination methods also to this these proofs. However, cut-elimination in presence of an induction rule is impossible in general [29]. On the other hand, a universal sequent $S\colon \Gamma \vdash \forall x.A(x)$, where x is supposed to range over the natural numbers, can be replaced by a sequence $S_n\colon A(\bar{n})_{n\in\mathbb{N}}$ for numerals $\bar{n}$. If instead of proving S by a single proof φ (using induction) we consider a proof sequence φ_n of S_n (like in the analysis of the Fürstenberg proof) we can use cut-elimination methods like `ceres` on the sequence. But that makes sense only if we succeed to describe the resolution refutations on the sequence of ccs's *uniformly* and (at least) obtain a uniform description of the sequence of corresponding Herbrand sequents. A general method of this type capable of handling Peano arithmetic is intrinsically very complex (in fact cut-elimination of this type proves the consistency of Peano arithmetic) and far outside of any means of automation. A partial solution of this problem for simple kind of inductions can be found in [14]. The elimination of lemmas in inductive proofs remains one of the major challenges of proof analysis.

7 Conclusion

We have presented a method of analyzing proofs via cut-elimination by resolution. As the core of the method consists of a theorem proving problem (the resolution refutation of a characteristic clause set) the real efficiency of the method is closely tied to that of automated theorem provers. Not only the problem of *finding* a proof of a theorem, but also the problem of cut-elimination of proofs, i.e. *finding proofs of a specific form* from existing proofs of a theorem can be a hard challenging problem. In theorem proving the main focus is on the production of *some proof* of a theorem (in a frequently unreadable form), less emphasis is laid on post-processing of the proof output and its interpretations by humans. Proofs like that of Fürstenberg cannot be obtained by automated theorem provers (due to the weak lemma structure of resolution and paramodulation in clause logic). We claim that the investigation of the relation between complex abstract proofs (using complex lemmas) and elementary proofs of a mathematical theorem may lead to deep insights into a mathematical theory, far beyond the knowledge that the theorem simply holds.

References

1. M. Aigner, G. M. Ziegler. Proofs from THE BOOK. Springer, 1998.

2. M. Baaz, A. Ciabattoni, C.G. Fermüller: Cut Elimination for First Order Gödel Logic by Hyperclause Resolution. Proc. of *LPAR'2008*. LNCS 5330, 451–466, 2008.
3. M. Baaz, S. Hetzl, A. Leitsch, C. Richter, H. Spohr: Proof Transformation by CERES. In: J.M. Borwein, W.M. Farmer, (eds.) MKM 2006, LNCS (LNAI), vol. 4108, pp. 82?93. Springer, Heidelberg, 2006.
4. M. Baaz, S. Hetzl, A. Leitsch, C. Richter, H. Spohr: CERES: An Analysis of Fürstenberg's Proof of the Infinity of Primes. *Theoretical Computer Science*, 403 (2–3), pp. 160-175, 2008.
5. M. Baaz, S. Hetzl, D. Weller: On the complexity of proof skolemization *Journal of Symbolic Logic*, 77(2), pp. 669-686, 2012.
6. M. Baaz, A. Leitsch: On skolemization and proof complexity. *Fundamenta Informaticae*, 20/4, pp. 353-379, 1994.
7. M. Baaz, A. Leitsch: Cut-Elimination and Redundancy-Elimination by Resolution. *Journal of Symbolic Computation*, 29, pp. 149-176, 2000.
8. M. Baaz, A. Leitsch: CERES in Many-Valued Logics. *Proceedings of LPAR'2005*, LNAI 3452, 1–20, 2005.
9. M. Baaz, A. Leitsch: Towards a Clausal Analysis of Cut-Elimination. *Journal of Symbolic Computation*, 41, pp. 381–410, 2006.
10. M. Baaz, A. Leitsch: Methods of Cut-Elimination. *Trends in Logic 34*. Springer, 2011.
11. U. Berger, W. Buchholz, H. Schwichtenberg: Refined Program Extraction from Classical Proofs. *Annals of Pure and Applied Logic*, 114(1-3), pp. 3–25, 2002.
12. U. Berger, S. Berghofer, P. Letouzey, H. Schwichtenberg: Program Extraction from Normalization Proofs. *Studia Logica*, 82(1), pp. 25–49, 2006.
13. W.S. Brianerd, L.H. Landweber: Theory of Computation. John Wiley & Sons, 1974.
14. C. Dunchev, A. Leitsch, M. Rukhaia, D. Weller: CERES for first-order schemata. CoRR abs/1303.4257 (2013).
15. H. Fürstenberg: On the infinitude of primes. *American Mathematical Monthly* 62, p. 353, 1955.
16. G. Gentzen: Untersuchungen über das logische Schließen. *Mathematische Zeitschrift* 39, pp. 405–431, 1934–1935.
17. J.Y. Girard: Proof Theory and Logical Complexity. in *Studies in Proof Theory*, Bibliopolis, Napoli, 1987.
18. K. Gödel: Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes. *Dialectica* 12, pp. 280–287, 1958.
19. S. Hetzl, A. Leitsch, D. Weller CERES in Higher-order Logic *Annals of Pure and Applied Logic* 162(12), pp. 1001–1034, 2011
20. S. Hetzl, A. Leitsch, D. Weller, B. Woltzenlogel Paleo: Herbrand Sequent Extraction. AISC/Calculemus/MKM 2008, LNAI 5144, pp. 462-477, 2008.
21. S. Hetzl, A. Leitsch, D. Weller, B. Woltzenlogel Paleo: Proof Analysis with HLK, CERES and ProofTool: Current Status and Future Directions. Proceedings of the CICM Workshop on Empirically Successful Automated Reasoning in Mathematics, CEUR Workshop Proceedings Vol-378 (2008), ISSN 1613-0073, 2008.
22. A. Leitsch: The Resolution Calculus. *EATCS Texts in Theoretical Computer Science*, Springer, Berlin, 1997.
23. A. Leitsch, G. Reis, B. Woltzenlogel Paleo: Towards CERes in intuitionistic logic. CSL 2012, pp. 485-499, 2012.
24. D. Miller: A Compact Representation of Proofs. *Studia Logica* 46/4, pp. 347–370, 1987.

25. V. P. Orevkov: Lower Bounds for Increasing Complexity of Derivations after Cut Elimination. *J. Soviet Mathematics*, pp. 2337–2350, 1982.
26. P. Pudlak: The lengths of proofs. In: *Handbook of Proof Theory*, S.R. Buss (ed), Elsevier 1998.
27. S.G. Simpson: Subsystems of Second Order Arithmetic. Springer, 1999.
28. R. Statman: Lower bounds on Herbrand's theorem. *Proc. of the Amer. Math. Soc.* 75, pp. 104–107, 1979.
29. G. Takeuti: Proof Theory. North-Holland, Amsterdam, 2nd edition, 1987.
30. A. Degtyarev and A. Voronkov: Equality Reasoning in Sequent-Based Calculi. *Handbook of Automated Reasoning* vol. I, ed. by A. Robinson and A. Voronkov, chapter 10, pp. 611-706, Elsevier Science, 2001.
31. B. Woltzenlogel Paleo: Herbrand Sequent Extraction. VDM Verlag Dr.Müller e.K. (February 7, 2008), ISBN-10: 3836461528.