1 Introduction[1]

This project addresses the *proof theory* of first-order logic extended by *Henkin quantifiers* and further extended to *independence-friendly logic*. Quantifiers in these logics restrict the dependency of variables from variables beyond the linear order and are named *branching quantifiers*.

Branching quantifiers were first introduced in a conference paper by Henkin [15] cf. [22] for an overview. Besides in mathematical logic, branching quantifiers appear in various contexts, such as natural languages [30], game theory [31] and computation [10].

Systems of partially ordered quantification are intermediate in strength between first-order logic and second-order logic. Similar to second-order logic, first-order logic extended by branching quantifiers is incomplete. In proof theory incomplete logics are represented by *partial proof systems*, c.f. the wealth of approaches dealing with partial proof systems for second-order logic. In an analytic setting, these partial systems allow the extraction of implicit information in proofs, i.e. proof mining.

However, in contrast to second-order logic only few results are dealing with the proof theoretic aspect of the use of branching quantifiers in partial systems. The project lies within this under-investigated area of research, focusing particularly on developing various calculi for branching quantifiers which admit cut-elimination. This is of central importance for the analysis of mathematical proofs involving structured objects and helpful for the investigation of linguistic argumentations. Moreover, the project aims at extending `CERES`, the up to date most efficient method for the analysis of first-order proofs, to the area of branching quantifiers.

The main tool to develop the intended calculi is to provide quantifier inference rules with generalized *eigenvariable conditions*. A more foundational aim of the project is therefore to characterize the logics obtainable from generalized eigenvariable conditions of classical first-order logic.

2 State of the art

2.1 Branching quantifiers in second-order logic

Henkin introduced the general idea of dependent quantifiers $Q_{m,n,F}$ that extends classical first-order logic. This leads to the notion of a partially ordered quantifier with m universal quantifiers and n existential quantifiers, where F is a function that determines

[1] This project is the resubmission of the project "Proof theory of Henkin quantifiers" (P 30173-N35) supported by the innovative comments and suggestions of the referees. The project title has been changed to emphasise the generality of the research and its contribution to `CERES`.

for each existential quantifier on which universal quantifiers it depends (m and n may be any finite number).

The simplest Henkin quantifier that is not definable in ordinary first-order logic is the quantifier Q_H binding four variables in a formula. A formula A using Q_H can be written as

$$A_H = \begin{pmatrix} \forall x & \exists u \\ \forall y & \exists v \end{pmatrix} A(x, y, u, v).$$

This is to be read "For every x there is a u and for every y there is a v (depending only on y)" such that $A(x, y, u, v)$. If the semantical meaning of this formula is given in second-order notation, the above formula is semantically equivalent to the second-order formula

$$\exists f \exists g \forall x \forall y A(x, y, f(x), g(y)),$$

where f and g are function variables. In general, any Henkin quantifier corresponds to a prefix of existential quantifiers on functions followed by universal quantifiers on objects. If the Henkin quantifier is balanced it may be written as

$$\begin{pmatrix} x_1^1 & ... & x_1^m & y_1^1 & ... & y_1^n \\ & ... & & & ... & \\ x_k^1 & ... & x_k^m & y_k^1 & ... & y_k^n \end{pmatrix}$$

where x_i^j are universal quantifiers, y_i^j are existential quantifiers and y_v^u depends only on x_v^1 ... x_v^m.

In [20] Hintikka and Sandu introduced independence-friendly logic (IF logic), which is a generalization of Henkin quantifiers to arbitrary quantifier dependencies. The syntax of IF logic is based on terms and atomic formulas defined exactly as in first-order logic. Formulas of IF logic are defined as follows:

1. Any atomic formula φ is an IF formula.
2. If φ is an IF formula, then $\neg\varphi$ is an IF formula.
3. If φ and ψ are IF formulas, then $\varphi \circ \psi$ for $\circ \in \{\wedge, \vee\}$ is an IF formula.
4. If φ is a formula, v is a variable and V is a finite set of variables, then $(\exists v/V)\varphi$ and $(\forall v/V)\varphi$ are IF formulas.

The set of free variables $FV(\varphi)$ of an IF formula φ is defined as in usual first-order logic and we add: $FV((\exists v/V)\varphi) = FV((\forall v/V)\varphi) = (FV(\varphi)\backslash\{v\}) \cup V$.

The meaning of an IF formula φ is defined by its translation to second-order logic. Free variables in φ are considered to be universally quantified and we define $U = \emptyset$.

Then the translation $T_U(\varphi)$ of an IF formula φ relative to U is inductively defined as follows:

1. $T_U(\varphi) = \varphi$ if φ is atomic.
2. $T_U(\varphi \circ \psi) = T_U(\varphi) \circ T_U(\psi)$ for $\circ \in \{\wedge, \vee\}$.
3. $T_U((\forall v/V)\varphi) = \forall v T_{U \cup v}(\varphi)$ if $(\forall v/V)\varphi$ occurs positively and $T_{U \cup v}(\varphi)\{v \leftarrow f_v(y_1, \ldots, y_n)\}$ if $(\forall v/V)\varphi$ occurs negatively, where f_v is a new function symbol and $y_1, \ldots, y_n$ is a list of the variables in $U \backslash V$.
4. $T_U((\exists v/V)\varphi) = T_{U \cup v}(\varphi)\{v \leftarrow f_v(y_1, \ldots, y_n)\}$ if $(\exists v/V)\varphi$ occurs positively, where f_v is a new function symbol and $y_1, \ldots, y_n$ is a list of the variables in $U \backslash V$ and $\exists v T_{U \cup v}(\varphi)$ if $(\exists v/V)\varphi$ occurs negatively.

Note that IF logic extends Henkin quantifiers, which can be considered as expression dominated by function variables of the same arity.

2.2 Fundamental properties

The first nontrivial theorem about branching quantifiers was formulated in [15] and proved by Ehrenfeucht, where he showed that the class of all finite models is definable by a sentence with branching quantifiers. Therefore classical logic containing Q_H is essentially stronger than usual first-order logic, i.e. the set of tautologies is not recursively enumerable.

It is known that classical logic extended by Q_H does not satisfy the compactness, Löwenheim-Skolem and interpolation theorems, also the Beth property fails [24] and [29]. In fact, this logic is very expressive [8] and [21]. However, it is shown that the downward Löwenheim-Skolem theorem holds for those sentences of first-order predicate calculus extended by Q_H whose only non-logical symbol is the equality symbol [24].

Already classical logic extended by Q_H is Σ^1_1-complete [13], [36].

2.3 Decidability questions

In [21] it is shown that the monadic fragment of classical logic extended by Q_H is decidable. The same logic with at least one binary function or at least one binary predicate added has a nonarithmetical set of tautologies [28]. (Only logics restricted to poor vocabularies - monadic or empty - admit an arithmetical degree of unsolvability [28]).

2.4 Results on proof theory: partial proof systems

As no first-order logic containing Q_H is recursively enumerable, only partial proof theoretic representations are possible. (Note that partial representations are not unusual in proof theory; there are many results about partial second-order systems in the literature.) However, there are only few papers that deal with a proof theory for branching quantifiers. The most important paper is the work of Lopez-Escobar [25]. The system of Lopez-Escobar is semantically related to interpretations in non-standard models; we discuss the system of Lopez-Escobar in more detail in Section 2.5. In Takeuti's Proof Theory §24 [35] an infinitary concept of generalized quantifiers is introduced where Henkin quantifiers can be accommodated. Note however that this approach is based on the assumption that there are no function symbols in the language.

2.5 Proof-theoretic prerequisites

Since the aim of this project is to develop a proof theoretic representation of branching quantifiers within classical logic, we prefer the format of sequent calculus admitting cut-elimination. In our opinion Lopez-Escobar [25] has chosen a natural deduction format based on the assumption that only the introduction rule for Q_H has to be formulated:

$$\frac{\begin{array}{c}\Gamma\\ \vdots\\ A(a,b,s(a),t(b))\end{array}}{\begin{pmatrix} \forall x & \exists u \\ \forall y & \exists v \end{pmatrix} A(x,y,u,v)}$$

where a and b fulfil the obvious eigenvariable conditions. The elimination rule is formulated in analogy to the elimination rule for existential quantifiers. We are convinced that the natural deduction format is less suitable for classical logic, but it may be worth to come back to the approach of Lopez-Escobar for the representation of branching quantifiers in intuitionistic logic.

2.5.1 Traditional methods of cut-elimination

Cut-elimination has been introduced by Gentzen in [14]. He proved in his famous Hauptsatz that the cut-rule can always be eliminated from formal proofs in systems like **LK**. Gentzen introduced a method for cut-elimination, which can be described as *reductive cut-elimination*. The characteristic feature of this method is a stepwise rewriting of the proof, based on a rewrite system $\mathcal{R}$. Cut-formulas are decomposed w.r.t. their outermost logical operator, leading to a reduction of their logical complexity (their *grade*). Moreover, cut-inferences are shifted upwards, leading to a reduction of the *rank* of the corresponding cut-formulas. Though reductive cut-elimination has a wide range of applications, Gentzen's method is algorithmically expensive. The reason is that single steps of the method are local and therefore independent of the global structure of the proof.

Another reductive method for cut-elimination is the method by Schütte and Tait [34]. It is similar to Gentzen's cut-elimination, where it reduces the grade, but it selects the cuts in a different way. While Gentzen's procedure selects an uppermost cut, Schütte-Tait's procedure selects a largest one (w.r.t. the logical complexity of the cut-formula).

2.5.2 Cut-elimination by resolution, `CERES`

Baaz and Leitsch introduced an alternative cut-elimination method based on resolution called `CERES` (cut elimination by resolution) [5]. The technique relies on the resolution method from automated-theorem proving. In contrast to reductive cut-elimination, which operates on small parts of the proof, `CERES` takes the global structure of an **LK**-proof into account. 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 given proof. Then compute 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) [7].

Originally `CERES` was developed for classical logic, but it has also been successfully extended to higher-order logic [16],non-classical logics [6], [2] and intuitionistic logic [23]. While first-order `CERES` proved efficient in the analysis of real mathematical proofs (e.g. in the analysis of Fürstenberg's proof of the infinitude of primes [3]) higher-order `CERES` remained of theoretical interest only: due to the undecidability of higher-order unification the refutation of the (higher-order) characteristic sequents sets becomes a very hard problem (only a few simple proofs could be analyzed). This indicates that first-order formulations of problems (whenever possible) will be beneficial to proof analysis via `CERES`.

2.6 Branching quantifiers in mathematical proofs

At first glance it is not immediate to see that branching quantifiers extend the usual tools of mathematical argumentation. They are replaced by structural features, such as vectors or matrices. In contrast to these structural features, branching quantifiers would enable to deal directly with the components of the structured object without encoding. For instance, using Q_H we may express directly a function with two-dimensional vectors as arguments without referring to the vectors as objects. Another example is the concept of uniform convergence, c.f. [22] introduction.

The avoidance of branching quantifiers is therefore connected to seemingly minor extensions of the mathematical language with corresponding additional axioms such as pairing and projections. However, when it comes to a proof theoretical analysis of mathematical arguments, such differences in the formalization have a considerable impact e.g. on structure and complexity of the Herbrand disjunction.

2.7 Branching quantifiers in linguistics

The main linguistic question concerning branching quantifiers is: "Are non-linear quantifiers needed for the representation of natural language?" This question has been formulated in [19] for the first time.

We refer to [30] and [33] for references on quantifiers in natural language: among the sentences emphasising the use of Henkin quantifiers are "Some relatives of each villager and some relatives of each townsmen hate each other.", "In my class, most boys and most girls dated each other.", "Most of the parliament members referred to each other." and "The richer the firm, the more powerful its CEO.".

3 Aims of the project

The main aim of this project is to develop suitable analytic calculi as foundation for a proof theory for branching quantifiers. Despite the inherent incompleteness w.r.t. the semantics of branching quantifiers these calculi should serve the purpose to prove (some of the) theorems and to analyze derivations of corresponding statements. In particular we want to investigate two approaches. The first one is based on a translation to second-order logic, resulting in a calculus $\mathbf{LK}_f$. This approach is immediate as formulas containing branching quantifiers can be understood as second-order formulas with function variables. But in case of applications of `CERES` we are dealing with second-order unification, which is undecidable [17], [37]. Therefore a second approach is needed, where we aim to develop a representation of branching quantifiers in the format of first-order logic. This novel calculus $\mathbf{LK}_h$ will be designed to be suitable for the intended applications. We have already mentioned in Section 2.5.2 that the method `CERES` performs best in first-order logic; hence the development of $\mathbf{LK}_h$ should lay the ground for an efficient proof analysis.

3.1 Representations in partial systems of second-order logic

As described, formulas containing branching quantifiers can be expressed in second-order logic [15], [10], [13]. For this representation function variables but no predicate variables are necessary. Consider for example the Henkin formula

$$A_H = \begin{pmatrix} \forall x & \exists u \\ \forall y & \exists v \end{pmatrix} A(x, y, u, v).$$

Its translation to second-order logic gives the formula

$$A^* \colon \exists f \exists g \forall x \forall y. A(x, y, f(x), g(y)).$$

In order to verify the formula A_H we give a proof of A^* in a suitable partial function calculus (Funktionenkalkül) which, besides the standard first-order features contains function variables of arbitrary arity. Surprisingly this fragment of second-order logic did not receive much attention.

The language of this function calculus $\mathbf{LK}_f$ is the language of $\mathbf{LK}$ extended by countably many free and bound function variables. The semantics is the semantics of $\mathbf{LK}$ extended by the semantics of the function variables, this means that universal quantification over some function variable X iterates through all terms and that existential quantification iterates over at least one term.

$\mathbf{LK}_f$ is $\mathbf{LK}$ extended by four quantifier rules:

- $\forall$-introduction for second-order function variables

$$\frac{A\{X \leftarrow \lambda\overline{x}.t\}, \Gamma \vdash \Delta}{(\forall X)A, \Gamma \vdash \Delta}\ \forall_l^f \qquad\qquad \frac{\Gamma \vdash \Delta, A\{X \leftarrow Y\}}{\Gamma \vdash \Delta, (\forall X)A}\ \forall_r^f$$

  where X is a second-order function variable, t is a term with free variables not bound in A (t may contain second-order variables) and Y is a second-order eigenvariable of same type as X.

- $\exists$-introduction for second-order function variables

$$\frac{A\{X \leftarrow Y\}, \Gamma \vdash \Delta}{(\exists X)A, \Gamma \vdash \Delta}\ \exists_l^f \qquad\qquad \frac{\Gamma \vdash \Delta, A\{X \leftarrow \lambda\overline{x}.t\}}{\Gamma \vdash \Delta, (\exists X)A}\ \exists_r^f$$

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

Though this logic is, of course, not complete it enjoys several nice semantic and proof theoretic features:

1. It is complete w.r.t. term models (this can be shown via a Schütte-type completeness proof) which define a very natural non-standard semantics. This means that existence in this calculus relates to definable functions.

2. Gentzen's and Schütte-Tait's cut-elimination can be carried over from **LK** without major changes and allows proof mining of functional interconnections.

3. The downward Löwenheim-Skolem theorem holds.

4. As also prenex forms are closed under substitution (which does not hold for full second-order logic) we also obtain a midsequent theorem and can extract Herbrand sequents from proofs.

The main open problem of $\mathbf{LK}_f$ concerns the use of Skolem functionals replacing strong function quantifiers. We intend to show that suitable chosen Skolem functionals are eliminable from Herbrand disjunctions using an approach similar to that of the Second Epsilon Theorem [18]. In addition we will try to establish suitable variants of the interpolation and Beth definability theorem using extensions of Maehara's theorem [35]. Note that this stands in contrast to properties of the full second-order system for Henkin quantifiers.

Henkin quantifiers formalized as above are weak in the sense that e.g. $(\forall x)(\exists x')P(x,x') \to \begin{pmatrix} \forall x & \exists x' \\ \forall y & \exists y' \end{pmatrix} P(x,x')$ is not derivable. This follows from the non-derivability of the Axiom of Choice $\forall x \exists y P(x,y) \to \exists f \forall x P(x, f(x))$ c.f. [22] page 249. Therefore, in our context the linear Henkin quantifier $\begin{pmatrix} \forall x & \exists y \end{pmatrix} P(x,y)$ is not equivalent to $\forall x \exists y P(x,y)$ (consequently, the systems LB and LS in [27] are stronger than our system of Henkin quantifiers).

As the emphasis of this project is the automated analysis of proofs and argumentations we prefer the explicit denotation of functional interrelations: for example, adding $\forall x \exists y P(x,y) \to \begin{pmatrix} \forall x & \exists y \end{pmatrix} P(x,y)$ to the antecedent of a sequent expresses the assumption that there is a choice function for P.

3.2 Analytic sequent calculi for branching quantifiers in the format of first-order logic

For simplicity we describe the intended general framework w.r.t. Q_H. One possible approach to establish a suitable calculus $\mathbf{LK}_h$ is the following. Consider the Henkin quantifier Q_H occurring in the formula

$$A_H = \begin{pmatrix} \forall x & \exists u \\ \forall y & \exists v \end{pmatrix} A(x,y,u,v)$$

as a macro of four inferences in $\mathbf{LK}_f$ deriving $A^*\colon \exists f \exists g \forall x \forall y. A(x, y, f(x), g(y))$ from a suitable premise and investigate the eigenvariable conditions. We suggest the formation of right and left inference rules for Q_H in the following way:

Right: The $\mathbf{LK}_f$ premise is $\Gamma \vdash \Delta, A(a, b, t_1, t_2)$, where a and b are eigenvariables not allowed to occur in the lower sequent and t_1 and t_2 are terms s.t. t_1 must not contain b and t_2 must not contain a. This leads to the obvious rule (already used by Lopez-Escobar [25]):

$$\frac{\Gamma \vdash \Delta, A(a, b, t_1, t_2)}{\Gamma \vdash \Delta, Q_H A(x, y, u, v)}\ Q_{H_r}$$

Obviously, the premise of this rule allows the derivation of the macro in $\mathbf{LK}_f$.

Left: The premise of the left inference rule of $\mathbf{LK}_f$ is $A(t_1, t_2, f(t_1), g(t_2)), \Pi \vdash \Gamma$ where f and g are eigenvariables. Replace $f(t_1)$ everywhere by a and $g(t_2)$ everywhere by b. The suggested inference rules are therefore:

$$\frac{A(t_1', t_2', a, b), \Pi \vdash \Gamma}{Q_H A(x, y, u, v), \Pi \vdash \Gamma}\ Q_{H_{l_1}}$$

a and b are eigenvariables not allowed to occur in the lower sequent and t_1', t_2' are terms s.t. a does not occur in t_2' and a and b do not occur in t_1'.

$$\frac{A(t_1', t_2', a, b), \Pi \vdash \Gamma}{Q_H A(x, y, u, v), \Pi \vdash \Gamma}\ Q_{H_{l_2}}$$

a and b are eigenvariables not allowed to occur in the lower sequent and t_1', t_2' are terms s.t. b does not occur in t_1' and a and b do not occur in t_2'.

The eigenvariable conditions ensure that $f(t_1')$ and $g(t_2')$ can be simultaneously substituted into a and b, respectively. Therefore the premise of the inference of the $\mathbf{LK}_f$ macro can be reconstructed.

Remark: In case there are no two-place function symbols in the language then there is only one left inference rule for Q_H.

To illustrate the intended calculus, consider the following simple example:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{A(a,b,c,d) \vdash A(a,b,c,d)}{A(a,b,c,d) \vdash \exists v A(a,b,c,v)}\ \exists_r}{A(a,b,c,d) \vdash \exists u \exists v A(a,b,u,v)}\ \exists_r}{\begin{pmatrix} \forall x & \exists u \\ \forall y & \exists v \end{pmatrix} A(x,y,u,v) \vdash \exists u \exists v A(a,b,u,v)}\ Q_{H_l}}{\begin{pmatrix} \forall x & \exists u \\ \forall y & \exists v \end{pmatrix} A(x,y,u,v) \vdash \forall y \exists u \exists v A(a,y,u,v)}\ \forall_r}{\begin{pmatrix} \forall x & \exists u \\ \forall y & \exists v \end{pmatrix} A(x,y,u,v) \vdash \forall x \forall y \exists u \exists v A(x,y,u,v)}\ \forall_r$$

The midsequent is $A(a, b, c, d) \vdash A(a, b, c, d)$.

This approach can be extended to IF logic by giving up the locality of eigenvariable conditions, c.f. [1]: terms inferring the weak quantifiers (negative universal and positive existential) may contain the eigenvariables of the strong (positive universal and negative existential) quantifiers these weak quantifiers do not depend upon.

The main objective is to prove the redundancy of the cut-rule for $\mathbf{LK}_h$, where it may turn out to be necessary to modify the eigenvariable conditions (as above). The following approaches seem to be the most promising ones:

1. Prove cut-elimination using Gentzen or Schütte-Tait style approaches.
2. Refine cut-elimination for $\mathbf{LK}_f$ by keeping the inferences for the defining formulas for Henkin quantifiers as macros. This approach might use `CERES` on a meta level, i.e. on the level of $\mathbf{LK}_f$.
3. Use rule permutations to reconstruct the inference macros from the cut-free $\mathbf{LK}_f$ derivations.
4. Show cut-free completeness for a suitable collection of non-standard models.

Corollaries to the eliminability of the cut-rule are Herbrand's theorem in the variant of the midsequent theorem and interpolation via Maehara's lemma.

The proof theoretic Skolemization of the end-sequent is obtained in classical logic by replacing eigenvariables in strong quantifier inferences ($\forall$ right, $\exists$ left) by suitable Skolem terms. The same strategy may be used to obtain Skolem forms from cut-free $\mathbf{LK}_h$-proofs. It will be however necessary to show that such Skolem functions can be replaced by quantifiers again, similar to [4].

3.3 The development of `CERES` for mathematical and linguistic applications

One of the main applications of the intended analyticity of $\mathbf{LK}_h$ and the possibility to remove Skolem functions from cut-free proofs is the formulation of a suitable `CERES`-method based on standard unification. This will allow the analysis of mathematical proofs with structured objects as mentioned in 2.6.

For the application of `CERES` to linguistics we intend to connect to a substantial project of Christian Retoré[2] intending to construct a platform for argumentations. More precisely, the platform should describe interventions in texts as follows:

[2]Christian Retoré (Faculté des Sciences - Université de Montpellier) suggested to connect the projects (Christian Retoré, E-mail communication). This line of research is also in accordance with the suggestions of the referees.

1. A part of the previous argument (the so-called *anchor* of the intervention) is selected.

2. The selected part is reformulated in own words to identify misunderstandings.

3. *Agree*, *disagree* or *don't understand* are selected.

4. An argument for the selected opinion is provided.

`CERES` should support this platform by extracting Herbrand disjunctions in suitable cases to determine minimal preconditions for contradictions.

3.4 The expressibility of generalized eigenvariable conditions

The eigenvariables in the first-order formulation of branching quantifiers are subjected to the following conditions:

1. They do not occur in the conclusion of the rule.

2. Their occurrence in specific terms in the inference premise is constrained. The only constraint is non-occurrence.

We intend to determine the expressibility and the proof-theoretic properties of quantifiers based on such eigenvariable conditions. This includes the question whether all quantifiers definable in $\mathbf{LK}_f$ can be represented in this way.

3.5 Epsilon calculus[3]

Epsilon calculus, introduced by David Hilbert, see [18, 26], replaces quantifiers in classical logic by ε-terms. $\exists x A(x)$ is translated into $A(\varepsilon_x A(x))$ and $\forall x A(x)$ is translated into $A(\varepsilon_x \neg A(x))$ or, if preferred, into $A(\tau_x A(x))$ (formulas are translated inside-out). A proof in first-order logic is translated into epsilon calculus by adding critical formulas $A(t) \supset A(\varepsilon_x A(x))$ and $A(\tau_x A(x)) \supset A(t)$ for weak quantifier inferences to classical propositional logic. Strong quantifiers inferences depending on eigenvariable conditions are replaced by substitutions. The first epsilon theorem is an elimination device for critical formulas by which purely existential statements are transformed into a Herbrand disjunction.

It is intended to extend the concept of epsilon calculus to our partial systems with branching quantifiers. The main principle is that in the translation of depending variables those variables they do not depend upon and the corresponding ε-terms are blocked.

[3]The inclusion of this paragraph has been suggested by the referees.

Example: To translate

$$\begin{pmatrix} \forall x & \exists u \\ \forall y & \exists v \end{pmatrix} A(x,y,u,v)$$

into epsilon calculus, we first translate the u/v components obtaining

$$A(a, b, \varepsilon_u A(a, \#, u, v), \varepsilon_v A(\#, b, u, v)).$$

The translation of the full formula results from substituting at first
$t(a) = \tau_y A(a, y, \varepsilon_u A(a, \#, u, v), \varepsilon_v A(\#, y, u, v))$ for b and then
$\tau_y A(a, t(a), \varepsilon_u A(a, \#, u, v), \varepsilon_v A(\#, t(a), u, v))$ for a. (Note that the order of the translation of the x, y components is proof theoretically irrelevant.)

We intend to extend the first epsilon theorem to these concepts. Note that the advantage of the epsilon formalism for branching quantifiers is that they can be handled locally.

3.6 Implementations and experiments

We intend to develop implementations of the most important algorithms resulting from this project. More precisely, we want to extend the existing implementations of the method `CERES` (see GAPT [12]) which works on proofs in ordinary **LK** to input proofs in $\mathbf{LK}_h$.

The clause forms resulting from the proof theoretic Skolemization methods sketched in the preceding sections can be handled by resolution based automated theorem provers, as Vampire [32]. Example: To prove

$$\begin{pmatrix} \forall x & \exists u \\ \forall y & \exists v \end{pmatrix} A(x,y,u,v) \vdash \forall x \forall y \exists u \exists v A(x,y,u,v)$$

(where $A(x, y, u, v)$ is atomic) we have to refute the formula

$$Q_H A(x,y,u,v) \wedge \neg \forall x \forall y \exists u \exists v A(x,y,u,v).$$

The intended Skolemization is:

$$\forall x \forall y A(x, y, f(x), g(y)) \wedge \forall u \forall v \neg A(r, s, u, v),$$

which leads to the clause form:

$$\{\{A(x, y, f(x), g(y))\}, \{\neg A(r, s, x', y')\}\}$$

which can be refuted by resolution:

$$\frac{\{A(x,y,f(x),g(y))\} \qquad \{\neg A(r,s,x',y')\}}{\emptyset}$$

The soundness of this approach depends on the conservativity of the proof theoretic Skolemization used.

Experiments will compare the efficiency of the described approaches to

1. resolution based theorem provers based on the clause form of the problem using the usual encoding of branching quantifiers in mathematics by structure operations such as lists,

2. higher-order automated theorem provers such as Leo-II and Satallax [9,11] applied to direct translations of branching quantifiers to second order expressions.

Natural source of examples are domains in mathematics with finitely structured objects such as problems in theories of linear transformations and in finitely dimensional affine geometry. Branching quantifiers allow to formulate the problems without reference to e.g. the concept of list on object level.

4 Ethical issues and broader effects

No ethical issues beyond those relevant to any research project have to be taken into account.

5 Work plan and personnel

5.1 Personnel information

1. Ao.Univ.Prof. Dr.phil. Matthias Baaz
   Institut für Diskrete Mathematik und Geometrie E104
   Technische Universität Wien
   Wieder Hauptstrasse 8-10, 1040 Wien

2. Univ.Prof. Dr.phil. Alexander Leitsch
   Institut für Computersprachen E185.2
   Technische Universität Wien
   Favoritenstrasse 9, 1040 Wien

3. Dr.techn. Giselle Machado Nogueira Reis
   Carnegie Mellon University Qatar
   Doha, Qatar

Faculté des Sciences - Université de Montpellier
Montpellier, France
Christian Retoré is head of an important institute for computational linguistics. He made his scientific career in proof theory and therefore his experience will be crucial to the investigations of the linguistic aspects of this project.

References

[1] Juan P Aguilera and Matthias Baaz. Unsound inferences make proofs shorter. *arXiv preprint arXiv:1608.07703*, 2016.

[2] Matthias Baaz, Agata Ciabattoni, and Christian 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] Matthias Baaz, Stefan Hetzl, Alexander Leitsch, Clemens Richter, and Hendrik Spohr. CERES: An analysis of Fürstenberg's proof of the infinity of primes. *Theoretical Computer Science*, 403:160–175, 2008.

[4] Matthias Baaz, Stefan Hetzl, and Daniel Weller. On the complexity of proof deskolemization. *The Journal of Symbolic Logic*, 77(02):669–686, 2012.

[5] Matthias Baaz and Alexander Leitsch. Cut-elimination and redundancy-elimination by resolution. *Journal of Symbolic Computation*, 29(2):149–177, 2000.

[6] Matthias Baaz and Alexander Leitsch. Ceres in many-valued logics. In *Logic for Programming, Artificial Intelligence, and Reasoning*, pages 1–20. Springer, 2005.

[7] Matthias Baaz and Alexander Leitsch. *Methods of Cut-elimination*, volume 34 of *Trends in Logic*. Springer, 2011.

[8] Jon Barwise. Some applications of Henkin quantifiers. *Israel journal of mathematics*, 25(1):47–63, 1976.

[9] Christoph Benzmüller, Nik Sultana, Lawrence C. Paulson, and Frank Theiss. The higher-order prover leo-ii. *J. Autom. Reasoning*, 55(4):389–404, 2015.

[10] Andreas Blass and Yuri Gurevich. Henkin quantifiers and complete problems. 1986.

[11] Chad E. Brown. Satallax: An automatic higher-order prover. In *Automated Reasoning - 6th International Joint Conference, IJCAR 2012, Manchester, UK, June 26-29, 2012. Proceedings*, pages 111–117, 2012.

[12] Gabriel Ebner, Stefan Hetzl, Giselle Reis, Martin Riener, Simon Wolfsteiner, and Sebastian Zivota. System description: Gapt 2.0. In *International Joint Conference on Automated Reasoning*, pages 293–301. Springer, 2016.

[13] Herbert B. Enderton. Finite partially-ordered quantifiers. *Mathematical Logic Quarterly*, 16(8):393–397, 1970.

[14] Gerhard Gentzen. Untersuchungen über das logische Schließen. *Mathematische Zeitschrift*, 39:176–210,405–431, 1934-35.

[15] L. Henkin. Some remarks on infinitely long formulas. In *Journal of Symbolic Logic*, volume 30, pages 167–183. Pergamon Press, 1961.

[16] Stefan Hetzl, Alexander Leitsch, and Daniel Weller. CERES in higher-order logic. *Annals of Pure and Applied Logic*, 162(12):1001–1034, 2011.

[17] Stefan Hetzl, Alexander Leitsch, and Daniel Weller. Ceres in higher-order logic. *Annals of Pure and Applied Logic*, 162(12):1001–1034, 2011.

[18] David Hilbert and Paul Bernays. Grundlagen der Mathematik. 2, 1939.

[19] Jaakko Hintikka. Quantifiers vs. quantification theory. *Linguistic inquiry*, pages 153–177, 1974.

[20] Jaakko Hintikka and Gabriel Sandu. Informational independence as a semantical phenomenon. *Studies in Logic and the Foundations of Mathematics*, 126:571–589, 1989.

[21] Michał Krynicki and Alistair H Lachlan. On the semantics of the Henkin quantifier. *The Journal of Symbolic Logic*, 44(02):184–200, 1979.

[22] Michał Krynicki and Marcin Mostowski. Henkin quantifiers. In *Quantifiers: logics, models and computation*, pages 193–262. Springer, 1995.

[23] Alexander Leitsch, Giselle Reis, and Bruno Woltzenlogel Paleo. Towards CERes in intuitionistic logic. In *Computer Science Logic (CSL'12) - 26th International Workshop/21st Annual Conference of the EACSL, CSL 2012, September 3-6, 2012, Fontainebleau, France*, pages 485–499, 2012.

[24] E.G.K. Lopez-Escobar. A non-interpolation theorem. *Journal of Symbolic Logic*, 40(3):457–458, 1975.

[25] E.G.K. Lopez-Escobar. Formalizing a non-linear Henkin quantifier. *Fundamenta Mathematicae*, 138(2):93–101, 1991.

[26] Georg Moser and Richard Zach. The epsilon calculus and Herbrand complexity. *Studia Logica*, 82(1):133–155, 2006.

[27] Marcin Mostowski. Quantifiers definable by second order means. In *Quantifiers: logics, models and computation*, pages 181–214. Springer, 1995.

[28] Marcin Mostowski and Konrad Zdanowski. Degrees of logics with Henkin quantifiers in poor vocabularies. *Archive for Mathematical Logic*, 43(5):691–702, 2004.

[29] John Paulos. Noncharacterizability of the syntax set. *The Journal of Symbolic Logic*, 41(02):368–372, 1976.

[30] Stanley Peters and Dag Westerståhl. *Quantifiers in language and logic.* Oxford University Press, 2006.

[31] Gary Peterson, John Reif, and Salman Azhar. Lower bounds for multiplayer noncooperative games of incomplete information. *Computers & Mathematics with Applications*, 41(7):957–992, 2001.

[32] Alexandre Riazanov and Andrei Voronkov. Vampire 1.1. In *International Joint Conference in Automated Reasoning*, pages 376–380. Springer, 2001.

[33] Jakub Szymanik. Quantifiers and cognition: Logical and computational perspectives. *Studies in Linguistics and Philosophy. Springer, forthcoming*, 2016.

[34] William W. Tait. Normal derivability in classical logic. In *The Syntax and Semantics of Infinitary Languages*, pages 204–236. Springer, 1968.

[35] Gaisi Takeuti. *Proof Theory.* Courier Dover Publications, 2013.

[36] Wilbur John Walkoe. Finite partially-ordered quantification. *The Journal of Symbolic Logic*, 35(04):535–555, 1970.

[37] Daniel Weller. *CERES in Higher-Order Logic.* PhD thesis, Vienna University of Technology, 2010.

List of abbreviations

CERES Cut-elimination by resolution

LK ”Logischer klassischer Kalkül”

$\mathrm{LK_f}$ Function calculus

$\mathrm{LK_h}$ Henkin calculus

Σ_1^1 existential second-order in analytic hierarchy