Descente Infinie
+
Deduction

Claus-Peter Wirth

Dept. of Computer Science, Universität des Saarlandes, D–66123 Saarbrücken, Germany

cp@ags.uni-sb.de
http://ags.uni-sb.de/~cp/welcome.html

Research Report 737/2000, FB Informatik, Univ. Dortmund

August 2, 2000

Definition "wellfoundedness" changed, Aug. 5, 2001
(in honor of Jörg H. Siekmann's 60th birthday)

Polarity of choice-conditions swapped (Definition 9.2) and *quasi*-existentiality added (Definition 5.2, Definition 9.4, Lemma 11.2(6), Lemma 12.3(6), Definition 13.4, &c.), Feb. 1, 2002

Overall improved, esp. Lemma A.2 and its consequences, Dec. 13, 2002

Final edition, Feb. 1, 2003

Abstract: Although induction is omnipresent, inductive theorem proving in the form of *descente infinie* has not yet been integrated into full first-order deductive calculi. We present such an integration for (possibly even higher-order) classical logic. This integration is based on lemma and induction hypothesis application for free variable sequent and tableau calculi. We discuss the appropriateness of these types of calculi for this integration. The deductive part of this integration requires the first combination of raising, explicit variable dependency representation, the liberalized δ-rule, and preservation of solutions.

Contents

1 Motivation

When *deductive* validity (i.e. validity in all models) is introduced to students it comes with some calculus which is complete for first-order logic. If this calculus happens to be a sequent or a tableau calculus including a Cut rule, the students can compare the formal proofs with the informal ones they are hopefully acquainted with. To our opinion, these calculi can mirror the human proof search process better than others. Although to know a complete calculus does not mean to know much about theorem proving, the interrelation of a human-oriented calculus and the informal proof search of the students will turn out to be fruitful for their later mathematical work. It is a pity that—while nearly all proofs of a working mathematician include induction—nothing comparable for *inductive* validity is offered to the students. Some may argue that this is generally impossible because not even the first-order theory of the Peano algebra of natural numbers is recursively enumerable. Nevertheless, there really is some general way in that many working mathematicians search for an informal proof, may it be inductive or not. The inductive version of this proof search method goes back to the ancient Greeks, was rediscovered under the name "*descente infinie ou indéfinie*" by Pierre de Fermat (1607?–1665), and today is sometimes called "minimal criminal" (popular) or "implicit induction" (in opposition to "explicit induction" found e.g. in Boyer & Moore (1979) and Walther (1994); for disambiguation and history of the notions of "explicit", "implicit", "inductionless induction", and "*descente infinie*", cf. Wirth (2003)). If you want to prove a conjecture, this method requires that you show, for each assumed counterexample of the conjecture, the existence of another counterexample of the conjecture that is strictly smaller in some wellfounded ordering. The working mathematician applies it in the following fashion.

> He starts with the conjecture and simplifies it in case analyses which can be described as steps in a sequent or tableau calculus with Cut. When he realizes that the goals become similar to a different instance of the conjecture, he applies the conjecture just like a lemma, but keeps in mind that he actually has applied some induction hypothesis. Finally, he searches for some wellfounded ordering in which all the instances of the conjecture that he has applied as induction hypotheses are smaller than the original conjecture itself.

Looking for a formal inductive calculus for mirroring this style of human inductive theorem proving (*ITP*), the "implicit induction" of Bachmair (1988) was a starting point because it included induction hypothesis application; but it was restricted to first-order universally quantified pure equations and was not human-oriented. In Wirth (1997) we have presented a human-oriented inductive calculus for first-order universally quantified clausal logic. In Kühler (2000)—implemented as the QuodLibet system—this calculus is extended with a necessary concretion for reasoning on the induction ordering and with a tactic-based concept for proof guidance that is intended to partially automate the construction of proofs.

Extending this approach to full first-order logic turned out to be more difficult than expected: The state-of-the-art free variable first-order tableau calculi were not suited for the integration of *descente infinie* because they confused the Herbrand universes with their Skolem functions and did not preserve solutions (i.e. closing substitutions) (like Prolog does), thereby destroying the wellfoundedness of *descente infinie*.

In this paper we show how deductive theorem proving should look like from the point of view of induction in the style of *descente infinie*. While many possible choices in the design of calculi for deductive theorem proving do not really matter because the alternatives are dual to each other, they do make an important difference when one wants to integrate *descente infinie*.

Nevertheless, even the researcher who is not interested in induction can learn something on *computation of solutions*, *lemma application*, and the details of the difference between the δ-rule and the *liberalized δ-rule* (δ^+). These details become obvious when the preservation of solutions (closing substitutions) is considered besides soundness.

The crucial step, however, in this paper is the integration of induction in the style of *descente infinie* into the framework of state-of-the-art deductive theorem proving. This is achieved by an inference method called *induction hypothesis application*, which—roughly speaking—differs from lemma application in producing an additional ordering goal. Even for the experts in ITP, the following aspects will be new: Tableau presentation, full first-order and higher-order formulas, and free (existential) variables.

While the details and technical means of this paper (like a new semantics for Hilbert's ε-terms and for formulas with three different kinds of free variables (existential, universal, ε-constrained)) may turn out to be useful also in other contexts, they originate from the hard search for a solution to the single problem that gives this paper its meaning:

> The first *integration* of the ancient ideas of deduction and *descente infinie* into a formal framework of practical relevance.

While the usual notions of *completeness* are irrelevant for this integration because we are interested not just in the mere existence of proofs but of proofs of a special intentional form; in order to go beyond a philosophical discussion it is necessary to formally prove its *soundness* with mathematical rigor, although the semantical means for this proof are very involved. The technical difficulty of the proofs could be reduced by choosing a different special representation for each single aspect, but we think that the concepts gain credit from the possibility to integrate them into a uniform framework. The framework—once accepted—does not need the complicated and philosophically doubtful (cf. e.g. Wittgenstein (1939)) semantical justifications anymore. Due to the simplicity of the framework itself and due to the experience with an implemented sub-system (cf. Kühler (2000)) we have reason to hope that first-order realizations of it do not provide more difficulty to students than calculi for first-order deduction.

The paper organizes as follows: After introducing the relevant notions in Sections 2 to 4, we explicate the kernel of our new approach in Sections 5 to 14. Then we illustrate the practical relevance with several simple examples, including the search for a lower bound for the Ackermann function (Section 18) and Newman's Lemma (Section 19). After concluding in Section 21 we append all the proofs and notes. Please do try not to read the notes on a first reading!

Leicht beieinander wohnen die Gedanken,	The ideas live together easily,
Doch hart im Raume ſtoßen ſich die Sachen.	But the realizations clash heavily.

Friedrich Schiller. *Wallensteins Tod, 2. Aufzug, 2. Auftritt.*

2 Orientation

In this section we describe the area of this paper. As the design of inference systems for proof search is the subject of this paper, we discuss our design goals carefully and conclude that sequent and tableau calculi are the best candidates for satisfying them.

2.1 Design Goals for Inductive Inference Systems

Based on experiences with more or less automated inductive theorem provers for classical logic, such as NQTHM, INKA, RRL, UNICOM, SPIKE, EXPANDER, &c., cf. Boyer & Moore (1988), Biundo &al. (1986), Kapur & Zhang (1989), Gramlich & Lindner (1991), Bouhoula & Rusinowitch (1995), Padawitz (1998), resp., we have come to adopt a rather pragmatic viewpoint with respect to ITP: Successful use of an inductive theorem prover in "real-life" problem domains has not been possible yet without a knowledgeable human user who can interact with the system on various levels. Accordingly, we think that even the development of the *theoretical* concepts of a new theorem prover—including its inference system—should begin with a clear emphasis on user interaction, whereas automatic proof guidance is seen as a long-term goal. Therefore, the following two requirements are main design goals for our inference systems:

I. We expect the inference system to comply with human (inductive) proof techniques in that it enables users to naturally realize their proof ideas in terms of the inference system.

II. Users should have no difficulties in understanding and searching for formal proofs represented with the inference system, no matter whether they try to follow them on the mere syntactical level or try to grasp overall "strategic" aspects of the proofs.

These design goals are contrary to those often found in deductive theorem proving, where most restrictive normal form calculi used to be preferred. Nevertheless, the point of view found in Giese (1998) is—though still more technical—already getting quite close to ours.

Refining the first design goal we obtain the following requirements:

I.1. *All proof problems and sub-problems, defining equations, lemmas, and induction hypotheses should be represented homogeneously*, so that the conclusion and all premises of any inference rule can be expressed in the same language. This enables the user to utilize the full power of the whole inference system on all problems and sub-problems and to choose freely and flexibly between eager or lazy strategies for any proof problem.

I.2. Another important point is that the inference system includes *inference rules for most elementary proof steps*, so that the user can force the prover to follow his proof ideas as closely as possible. We consider availability of atomic inference steps to be more important than having (derived) higher-level inference rules. Applied in non-trivial proof problems higher-level inference rules tend to be too restrictive, while an inference system comprising a multitude of simple "fine-grain" inference rules can be used to (interactively) construct even very difficult proofs.

Refining the second design goal we obtain the following requirements:

II.1. The inference system should support a *natural flow of information* in the sense that a certain decision can be delayed or a commitment deferred until the state of the proof attempt provides information that is sufficient to make a successful decision. Examples for unnatural flow of information are:

   (a) Instantiating induction hypotheses in induction step formulas of explicit induction long before the hypotheses become applicable, cf. Protzen (1994).

   (b) The γ-rule of sequent or tableau calculi without free variables, where instantiations have to be guessed long before it can be recognized which instantiations will make a proof attempt successful.

   (c) The $\vee$-introduction ($\vee$I) and the indirect-proof ($\perp_{\mathbf{c}}$ or $A \vee \neg A$) rules in natural deduction calculi for classical logic: The first requires a decision for one of two disjunctive alternatives and the second a decision of when to start an indirect proof, both of which are not at all necessary in classical logic.[1]

   Together with the homogeneous and flexible formulation of the inference rules, the natural flow of information should make it possible to replace a certain amount of proof planning based on some special abstractive representation (cf. e.g. Kerber (1998)) with heuristic search based on the homogeneous representation in the inference system.

II.2. Another requirement that is very important for efficiency of automated as well as interactive theorem proving is *goal-directedness*. Goal-directedness means that every problem in the graph of a proof attempt is connected with the theorem to be proved. For *inductive* theorem proving this is even more important than for *deductive* theorem proving: The crucial point is that we often have to invent some new lemmas to close the gap between the induction conclusion and the induction hypotheses. This creative invention can be guided by the user's knowledge of the domain or more automatically by the applicable lemmas, the (expanded) induction conclusion, and the induction hypotheses. Without the goal-directedness given by the connection with the induction conclusion *and* the induction hypotheses the missing lemmas can hardly be guessed. Note that (disregarding proof length) such creative steps are not necessary for *deductive* theorem proving because according to Gentzen's Hauptsatz a proof of a deductive theorem does not need to invent new formulas but can be restricted to "sub"-formulas of the theorem, although this notion of "sub"-formula is closed under instantiations with a usually infinite number of terms. On the contrary, the application of hypotheses and lemmas inside an inductive reasoning cycle cannot generally be eliminated in the form of Gentzen's Hauptsatz, cf. Kreisel (1965). Thus, for ITP, creativity cannot be restricted to finding the proper instantiations, but requires the invention of new formulas, possibly in an enrichment of the signature and in an extension of the specification.

As a first step towards achieving the above design goals, we have an inference system in mind that explicitly provides the concepts of *induction hypothesis* and *induction ordering*. With the latter concept we associate means of supplying induction ordering conditions with sufficient expressiveness and flexibility, i.e. explicit *weights*. Concerning the concept of induction hypothesis, we intend an inference system that does not "hide" several (applications of) induction hypotheses in a single inference step. We rather think of an inference system that "knows" what an induction hypothesis is, i.e. it includes inference rules that provide or apply induction hypotheses, given that certain ordering conditions resulting from these applications can be met by an induction ordering. Obviously, such an inference system is an inference system for *descente infinie* and not an inference system restricted to explicit induction as found in the ITP systems NQTHM, INKA, and RRL and still in Kreitz & Pientka (2000). Furthermore, the intended inference system supports eager as well as lazy generation of induction hypotheses and mutual induction. As a consequence, we obtain the following essential requirement for our inference system:

> The inference system must be capable of representing an induction hypothesis as a whole and in recognizable form. Not an inference rule nor input normalization may decompose a conjectured inductive theorem (into "sub"-formulas) before the induction hypotheses have been extracted from it.

2.2 Why Sequent and Tableau Calculi

This section requires some deeper understanding of inference systems and can well be skipped.

The considerations of Section 2.1 may help settling the following question: *Which deductive inference systems are well suited for an integration of descente infinie?* Note that in the more restricted framework of explicit induction, obtaining an inference system for ITP from a deductive inference system is far simpler: Since induction is captured in a single inference rule then, this "induction rule" can be just added to the given deductive inference system without affecting the rest of the inference system. When integrating *descente infinie*, however, the whole inference system is affected.

As proof search is very hard in *Hilbert calculi*, they are not adequate for theorem proving in general. Not too much better for proof search in classical logic are *natural deduction calculi*, cf. Gentzen (1935), Prawitz (1965), due to their unnatural flow of information as indicated in Section 2.1 (II.1.c). For *descente infinie*, natural deduction is additionally problematic because the proofs are augmented with assumptions that conflict with our concept of induction hypothesis.

Sergey Yu. Maslov's general idea of inversion (cf. Maslov (1971)) seems not to be appropriate for *inductive* theorem proving because an inductive proof where the induction conclusion is a formula at the top lacks goal-directedness when starting from the bottom. This lack of goal-directedness is very similar to that of the more familiar framework of *non-refutational resolution and paramodulation* (cf. Lee (1967)), where the resolution rule is applied to axioms only, and the task is to infer a formula that subsumes the theorem. Non-refutational resolution seems not very appropriate for *deductive* theorem proving because it is not goal-directed. In the context of *inductive* theorem proving, however, non-refutational resolution could be considered goal-directed because it starts with the axioms *plus the induction hypotheses*, and formulas that subsume the induction conclusions are to be inferred. Nevertheless, the goal-directedness given by the induction hypotheses is not sufficient. Since numerous lemmas may be applicable and applications of lemmas tend to be "closer" to applications of induction hypotheses than to the conclusions, it is practically impossible to find the appropriate ones unless the conclusion has been expanded to a large degree. Mostly under the name "rippling" a lot of work has been done on the heuristic control of this search problem, cf. e.g. Bundy &al. (1993), Hutter (1997). Furthermore, non-refutational resolution is not goal-directed in the base cases and in those parts of the proof that lie below applications of induction hypotheses. Finally, since inductive proofs use to follow the recursive definitions of the specification, non-refutational resolution requires to paramodulate with the defining rules from right to left. This can result in a high branching degree for some of the non-recursive cases and make a proper combination of the cases of the definition difficult (i.e. "unfolding" (or "expanding") a definition is easier than "folding" it). All in all, we conclude that non-refutational resolution and paramodulation as well as the general idea of inversion are not adequate for ITP. Thus, a reasonable integration of *descente infinie* into resolution and paramodulation must be refutational. Due to the problems discussed above, such an integration must be similar to EXPANDER (cf. Padawitz (1996), Padawitz (1998)), which is the only system for resolution and *descente infinie* I know. In EXPANDER, the induction hypotheses (which are special super-clauses (i.e. disjunctions of super-literals, which are conjunctions of literals) with additional existential variables) are applied very similar to the super-clauses in *Sergey Yu. Maslov's inverse method* (cf. Lifschitz (1989)) that generate inference rules operating on clauses. Moreover, goal-directedness also w.r.t. the induction conclusion is achieved here by starting from the negated induction conclusion in the form of a set of "goals", i.e. clauses in dual notation for user readability. Contrary to this, the inverse method starts from the set of tautologies, which has the advantage of deductive completeness but lacks goal-directedness w.r.t. the induction conclusion. Nevertheless, from my experiences with EXPANDER, I am not at all convinced that it satisfies our main design goals (I) and (II) of Section 2.1 well.

2.3 Sequent and Tableau Calculi

Now the only remaining family of well-known deductive inference systems is that of *sequent* (cf. Gentzen (1935), Lifschitz (1971)), *tableau* (cf. Smullyan (1968), Fitting (1996)), and *matrix calculi* (cf. Wallen (1990)). While matrix calculi have implementational advantages (cf. Section 20.2 and Wallen (1990)), in this paper we will only consider sequent and tableau calculi because their presentation is simpler. Fortunately, these calculi—cf. the following sections—do admit an integration of *descente infinie* in a way that is adequate in our opinion:

- By adding a powerful inference rule for applying induction hypotheses, we obtain an inference system for ITP that is optimally goal-directed w.r.t. the induction conclusions and sufficiently goal-directed w.r.t. the induction hypotheses.

- Starting with the sequent to be proved, the problem of proving a sequent (*goal*) is reduced to the problem of proving another set of sequents (*sub-goals*), which can be considered to be its children. Applying such reduction steps recursively results in a tree-like proof structure, which is augmented with cyclic arguments resulting from inference rules generated by induction hypotheses. A proof having the simple structure of a tree is *easy to understand* for users and a good basis for automatic tactics and heuristic proof search.

We assume some familiarity with sequent or tableau theorem proving. We recommend Gentzen (1935) and Fitting (1996) as excellent introductions to the subject.

In Smullyan (1968), rules for analytic deductive theorem proving are classified as α-, β-, γ-, and δ-rules independently from a concrete calculus.[2]

α-rules describe the simple and

β-rules the case-splitting (or branching) propositional proof steps.

γ-rules show existential properties, either by exhibiting a term witnessing to the existence or else by introducing a special kind of variable, called "dummy" in Prawitz (1960) and Kanger (1963), "free variable" in footnote 11 of Prawitz (1960) and in Fitting (1996), and "meta variable" in Giese (1998) e.g.. We will call these variables *free existential variables*. By the use of free existential variables we can delay the choice of a witnessing term until the state of the proof attempt gives us more information which choice is likely to result in a successful proof. It is the important addition of free existential variables that makes the major difference between the free variable calculi of Fitting (1996) and the calculi of Smullyan (1968). Since there use to be infinitely many possibly witnessing terms (and different branches may need different ones), the γ-rules (under assistance of the β-rules) often destroy the possibility to decide validity because they enable infinitely many γ-rule applications to the same formula.

δ-rules show universal properties simply with the help of a new symbol, called a "parameter" or an "eigenvariable", about which nothing is known. Since the present free existential variables must not be instantiated with this new parameter, in the standard framework of Skolemization and unification the parameter is given the present free existential variables as arguments. In this paper, however, we will use nullary parameters, which we call *free universal variables*. These variables are not free in the sense that terms to replace them may be chosen freely, but in the sense that their occurrences must not be bound by any quantifier or other binder. Our free universal variables are similar to the parameters of Kanger (1963) because a free existential variable may not be instantiated with all of them. We will store the information on the dependency between free existential variables and free universal variables in *variable-conditions*.

Other rules may be added for an appropriate treatment of often used reasoning like unification, rewriting with equalities and logical equivalences, and human-oriented reasoning like Cut.

Besides these rules we would like to have rules for lemma application that apply the theorem proved in one tree as a lemma in the proof tree of another theorem. Moreover, induction hypothesis application will look like lemma application but generate extra branches (in the latter tree), which require to prove that the instance of the applied theorem (induction hypothesis) is somehow smaller in a wellfounded ordering than the theorem (induction conclusion) that it is going to prove.

Although the following inference rules from Theor. 14.1 of Section 14 cannot be completely understood at this early stage, they may clear away some fog. They are presented in the sequent calculus style. Moreover, note that in old times when trees grew upwards, Gerhard Gentzen would have written the inference rules such that passing the line meant consequence. We have inverted the rules. Thus, in our case, passing the line means reduction, and trees grow downwards.

Let A and B be formulas, $\mathit{\Gamma}$, $\mathit{\Pi}$, and $\mathit{\Lambda}$ be sequents (i.e. disjunctive lists of formulas), $x \in \mathrm{V}_{\mathrm{bound}}$ a bound variable, and $\mathcal{F}$ the current proof forest containing all free variables already used, esp. those from A and $\mathit{\Gamma\Pi}$:

α-rules: $\dfrac{\mathit{\Gamma}\ \neg\neg A\ \mathit{\Pi}}{A\ \mathit{\Gamma}\ \mathit{\Pi}}$ $\quad\dfrac{\mathit{\Gamma}\ (A{\vee}B)\ \mathit{\Pi}}{A\ B\ \mathit{\Gamma}\ \mathit{\Pi}}$ $\quad\dfrac{\mathit{\Gamma}\ \neg(A{\wedge}B)\ \mathit{\Pi}}{\overline{A}\ \overline{B}\ \mathit{\Gamma}\ \mathit{\Pi}}$ $\quad\dfrac{\mathit{\Gamma}\ (A{\Rightarrow}B)\ \mathit{\Pi}}{\overline{A}\ B\ \mathit{\Gamma}\ \mathit{\Pi}}$ $\quad\dfrac{\mathit{\Gamma}\ (A{\Leftarrow}B)\ \mathit{\Pi}}{A\ \overline{B}\ \mathit{\Gamma}\ \mathit{\Pi}}$

β-rules: In the following rules we may choose none or one, but not both of the formulas in optional brackets!

$$\dfrac{\mathit{\Gamma}\ (A{\wedge}B)\ \mathit{\Pi}}{A\ \big[\,\overline{B}\,\big]\ \mathit{\Gamma}\ \mathit{\Pi}\qquad B\ \big[\,\overline{A}\,\big]\ \mathit{\Gamma}\ \mathit{\Pi}}\qquad\qquad \dfrac{\mathit{\Gamma}\ \neg(A{\vee}B)\ \mathit{\Pi}}{\overline{A}\ \big[\,B\,\big]\ \mathit{\Gamma}\ \mathit{\Pi}\qquad \overline{B}\ \big[\,A\,\big]\ \mathit{\Gamma}\ \mathit{\Pi}}$$

$$\dfrac{\mathit{\Gamma}\ \neg(A{\Rightarrow}B)\ \mathit{\Pi}}{A\ \big[\,B\,\big]\ \mathit{\Gamma}\ \mathit{\Pi}\qquad \overline{B}\ \big[\,\overline{A}\,\big]\ \mathit{\Gamma}\ \mathit{\Pi}}\qquad\qquad \dfrac{\mathit{\Gamma}\ \neg(A{\Leftarrow}B)\ \mathit{\Pi}}{\overline{A}\ \big[\,\overline{B}\,\big]\ \mathit{\Gamma}\ \mathit{\Pi}\qquad B\ \big[\,A\,\big]\ \mathit{\Gamma}\ \mathit{\Pi}}$$

γ-rules: Let $x^{\exists} \in \mathrm{V}_{\exists} \setminus \mathcal{V}(\mathcal{F})$ be a new[3] free existential variable:

$$\dfrac{\mathit{\Gamma}\quad \exists x.\ A\quad \mathit{\Pi}}{A\{x{\mapsto}x^{\exists}\}\quad \mathit{\Gamma}\quad \exists x.\ A\quad \mathit{\Pi}}\qquad\qquad \dfrac{\mathit{\Gamma}\quad \neg\forall x.\ A\quad \mathit{\Pi}}{\overline{A\{x{\mapsto}x^{\exists}\}}\quad \mathit{\Gamma}\quad \neg\forall x.\ A\quad \mathit{\Pi}}$$

δ-rules: Let $x^{\forall,\mathrm{w}} \in \mathrm{V}_{\forall,\mathrm{w}} \setminus \mathcal{V}(\mathcal{F})$ be a new[4] weak free universal variable:

$$\dfrac{\mathit{\Gamma}\quad \forall x.\ A\quad \mathit{\Pi}}{A\{x{\mapsto}x^{\forall,\mathrm{w}}\}\quad \mathit{\Gamma}\quad \mathit{\Pi}}\ \Big(\ \mathcal{V}_{\exists}(A, \mathit{\Gamma\Pi}, \sqsupset) \cup \mathcal{V}_{\forall,\mathrm{s}}(A, \mathit{\Gamma\Pi}, \sqsupset)\ \Big) \times \{x^{\forall,\mathrm{w}}\}$$

$$\dfrac{\mathit{\Gamma}\quad \neg\exists x.\ A\quad \mathit{\Pi}}{\overline{A\{x{\mapsto}x^{\forall,\mathrm{w}}\}}\quad \mathit{\Gamma}\quad \mathit{\Pi}}\ \Big(\ \mathcal{V}_{\exists}(A, \mathit{\Gamma\Pi}, \sqsupset) \cup \mathcal{V}_{\forall,\mathrm{s}}(A, \mathit{\Gamma\Pi}, \sqsupset)\ \Big) \times \{x^{\forall,\mathrm{w}}\}$$

Liberalized δ-rules: Let $x^{\forall,\mathrm{s}} \in \mathrm{V}_{\forall,\mathrm{s}} \setminus \mathcal{V}(\mathcal{F})$ be a new[5] strong free universal variable:

$$\dfrac{\mathit{\Gamma}\quad \forall x.\ A\quad \mathit{\Pi}}{A\{x{\mapsto}x^{\forall,\mathrm{s}}\}\quad \mathit{\Gamma}\quad \mathit{\Pi}}\ \begin{array}{l}\{(x^{\forall,\mathrm{s}}, \overline{A\{x{\mapsto}x^{\forall,\mathrm{s}}\}})\}\\ \mathcal{V}_{\mathrm{free}}(A) \times \{x^{\forall,\mathrm{s}}\}\end{array}$$

$$\dfrac{\mathit{\Gamma}\quad \neg\exists x.\ A\quad \mathit{\Pi}}{\overline{A\{x{\mapsto}x^{\forall,\mathrm{s}}\}}\quad \mathit{\Gamma}\quad \mathit{\Pi}}\ \begin{array}{l}\{(x^{\forall,\mathrm{s}}, A\{x{\mapsto}x^{\forall,\mathrm{s}}\})\}\\ \mathcal{V}_{\mathrm{free}}(A) \times \{x^{\forall,\mathrm{s}}\}\end{array}$$

Rewrite-Rules: Let s and t be terms (of the same type). Let B be one of the formulas $(s{\neq}t)$ or $(t{\neq}s)$. Let $A[t]$ denote the formula $A[s]$ with some occurrences of s replaced with t:

$$\dfrac{\mathit{\Gamma}\ \ A[s]\ \ \mathit{\Pi}\ \ B\ \ \mathit{\Lambda}}{A[t]\ \ \mathit{\Gamma}\ \ \mathit{\Pi}\ \ B\ \ \mathit{\Lambda}}\qquad\qquad \dfrac{\mathit{\Gamma}\ \ B\ \ \mathit{\Pi}\ \ A[s]\ \ \mathit{\Lambda}}{A[t]\ \ \mathit{\Gamma}\ \ B\ \ \mathit{\Pi}\ \ \mathit{\Lambda}}$$

Cut:

$$\dfrac{\mathit{\Gamma}}{A\ \ \mathit{\Gamma}\qquad \overline{A}\ \ \mathit{\Gamma}}$$

3 Basics

We make use of "[...]" for stating two definitions, lemmas, or theorems (and their proofs &c.) in one, where the parts between '[' and ']' are optional and are meant to be all included or all omitted. '$\mathbf{N}$' denotes the set of and '$\prec$' the ordering on natural numbers. '$\mathbf{Z}$' denotes the set of integers. We define $\mathbf{N}_+ := \{\, n \in \mathbf{N} \mid 0 \neq n \,\}$. We use '$\uplus$' for the union of disjoint classes and 'id' for the identity function. For classes R, A, and B we define *domain* and *range*, (domain-) *restriction* and *range-restriction*, as well as *image*[6] and *reverse-image*:

$$\begin{array}{lclclcl} \mathrm{dom}(R) & := & \{\, a \mid \exists b.\ (a,b) \in R \,\} & ; & \mathrm{ran}(R) & := & \{\, b \mid \exists a.\ (a,b) \in R \,\} \quad ; \\ {}_A{\upharpoonleft} R & := & \{\, (a,b) \in R \mid a \in A \,\} & ; & R{\upharpoonright}_B & := & \{\, (a,b) \in R \mid b \in B \,\} \quad ; \\ \langle A \rangle R & := & \{\, b \mid \exists a \in A.\ (a,b) \in R \,\}; & & R\langle B \rangle & := & \{\, a \mid \exists b \in B.\ (a,b) \in R \,\}. \end{array}$$

Let 'R' denote a binary relation. R is said to be a relation *on* A if $\mathrm{dom}(R) \cup \mathrm{ran}(R) \subseteq A$. R is *irreflexive* if $\mathrm{id} \cap R = \emptyset$. It is *$A$-reflexive* if ${}_A{\upharpoonleft}\mathrm{id} \subseteq R$. Simply speaking of a *reflexive* relation we refer to the biggest A that is appropriate in the local context, and referring to this A we write R^0 to ambiguously denote ${}_A{\upharpoonleft}\mathrm{id}$. Furthermore, we write R^1 to denote R. For $n \in \mathbf{N}_+$ we write R^{n+1} to denote $R^n \circ R$, so that R^n denotes the n step relation for R. The *transitive closure* of R is $R^+ := \bigcup_{n \in \mathbf{N}_+} R^n$. The *reflexive & transitive closure* of R is $R^* := \bigcup_{n \in \mathbf{N}} R^n$. The *reverse*[7] of R will be denoted with R^{-1}. A sequence $(s_i)_{i \in \mathbf{N}}$ is *non-terminating in* R if $s_i \mathrel{R} s_{i+1}$ for all $i \in \mathbf{N}$. R is *terminating* if there are no non-terminating sequences in R. A relation R (on A) is *wellfounded* if any non-empty class B ($\subseteq A$) has an R-minimal element, i.e. $\exists a \in B.\ \neg\exists a' \in B.\ a' \mathrel{R} a$.

A *quasi-ordering* '$\lesssim$' *on* a class A is an A-reflexive and transitive (binary) relation on A. As with all our asymmetric relation symbols we define $a \gtrsim b$ if $b \lesssim a$. By an (irreflexive) *ordering* '$<$' we mean an irreflexive and transitive relation, sometimes called "strict partial ordering" &c. by other authors. A *reflexive ordering* '$\leq$' *on* A is an A-reflexive, antisymmetric, and transitive relation on A. The *ordering* $<$ *of a quasi-ordering or a reflexive ordering* $\lesssim$ is $\lesssim \setminus \gtrsim$, and $\lesssim$ is called *wellfounded* if $<$ is wellfounded.

Furthermore, we use '$\emptyset$' to denote the empty set as well as the empty function or empty word. Functions are nothing but (right-) unique relations and the meaning of '$f \circ g$' does not depend on f and g being only relations or even functions. Thus, '$(f \circ g)(x)$' means '$g(f(x))$'. The *class of total functions from A to B* is denoted with $A \to B$. The *class of (possibly) partial functions from A to B* is denoted with $A \rightsquigarrow B$. Both $\to$ and $\rightsquigarrow$ associate to the right, i.e. $A \rightsquigarrow B \to C$ reads $A \rightsquigarrow (B \to C)$.

Lemma 3.1 *If R is a wellfounded relation, then R^+ is a wellfounded ordering.*

3.1 Syntactical Conventions

We define a *sequent* to be a list of formulas.[8] The *conjugate* of a formula A (written: $\overline{A}$) is the formula B if A is of the form $\neg B$, and the formula $\neg A$ otherwise. Note that the conjugate of the conjugate of a formula is the original formula A again, unless A has the form $\neg\neg B$.

In the tradition of Gentzen (1935), Hilbert & Bernays (1968/70), and Snyder & Gallier (1989) we assume the following four sets of symbols to be disjoint:

$\mathrm{V}_{\exists}$	*free existential variables*, i.e. the free variables of Fitting (1996)
$\mathrm{V}_{\forall}$	*free universal variables*, i.e. nullary parameters, instead of Skolem functions
$\mathrm{V}_{\mathrm{bound}}$	*bound variables*, i.e. variables for bound use only
Σ	*constants*, i.e. the function (and predicate) symbols from the signature

We split the free universal variables $\mathrm{V}_{\forall}$ into *weak free universal variables* $\mathrm{V}_{\forall,\mathrm{w}}$ that behave as in the weak version of Wirth (1998) and are introduced by the non-liberalized δ-rules; and *strong free universal variables* $\mathrm{V}_{\forall,\mathrm{s}}$ that behave as in the strong version and are introduced by the liberalized δ-rules:

$$\mathrm{V}_{\forall} = \mathrm{V}_{\forall,\mathrm{w}} \uplus \mathrm{V}_{\forall,\mathrm{s}}.$$

We define the *free variables* by

$$\mathrm{V}_{\mathrm{free}} := \mathrm{V}_{\exists} \uplus \mathrm{V}_{\forall}$$

and the *variables* by

$$\mathrm{V} := \mathrm{V}_{\mathrm{bound}} \uplus \mathrm{V}_{\mathrm{free}}$$

We use '$\mathcal{V}_k(\Gamma)$' to denote the set of variables from V_k occurring in Γ.

Due to the possibility to rename bound variable occurrences w.l.o.g. ($\lambda\alpha$-conversion) and in the tradition of Hilbert & Bernays (1968/70), we do not permit binding of variables that already occur bound in a term or formula; i.e. e.g. $\forall x.\ A$ is only a formula in our sense if A does not contain a binder on x like "$\forall x.$ ", "$\exists x.$ ", "$\lambda x.$ ", "$\varepsilon x.$ ". The simple effect is that human beings can more easily read our formulas and that our γ- and δ-rules (and our $\lambda\beta$-reduction) can simply replace *all* occurrences of x. Moreover, we assume that all binders have minimal scope, e.g. $\forall x, y.\ A \wedge B$ reads $(\forall x.\ \forall y.\ A) \wedge B$.

For a substitution σ we denote with '$\Gamma\sigma$' the result of replacing each occurrence of a variable $x \in \mathrm{dom}(\sigma)$ in Γ with $\sigma(x)$. In default situations, we tacitly assume that all occurrences of variables from $\mathrm{V}_{\mathrm{bound}}$ in terms and formulas on top level and in the ranges of substitutions are *bound occurrences* (i.e. that a variable $x \in \mathrm{V}_{\mathrm{bound}}$ occurs only in the scope of a binder on x) and that each substitution σ satisfies $\mathrm{dom}(\sigma) \subseteq \mathrm{V}_{\mathrm{free}}$, so that no bound occurrences of variables can be replaced and no additional variable occurrences can become bound (i.e. captured) when applying σ.

While we use upper case Greek letters for sequences, we denote our weight constructs (which guarantee the wellfoundedness of induction) with Hebrew letters. Note that the beginning of the Hebrew alphabet is all we need: $\aleph$ aleph, $\beth$ beth, $\gimel$ gimel (which we do not use anymore because it was found hard to disambiguate it from $\beth$), and $\daleth$ daleth. Together with a sequent, a weight forms a *syntactical construct*[9]:

Definition 3.2 (Syntactical Construct, Weight Construct)
A *syntactical construct* is a pair $(\Gamma, \aleph)$ consisting of a sequent Γ and a weight construct $\aleph$. A *weight construct* is a triple $(w, <, \lesssim)$ consisting a ("weight") term w (of type say α), an ("induction ordering") term $<$, and an ("induction quasi-ordering") term $\lesssim$, both of type $\alpha \to \alpha \to \mathsf{bool}$ (or equivalently $\alpha{\times}\alpha$).
The set of all syntactical constructs is denoted by 'SynCons'. The function 'logic' extracts the *logic part* (here: the sequents) of a set G of syntactical constructs: $\mathrm{logic}(G) := \mathrm{dom}(G)$.

Syntactical constructs are the basic data structure for ITP, just as sequents or formulas are for the deductive case. They consist of purely syntactical elements although the terms w, $<$, $\lesssim$ may exceed the terms of our formulas and sequents. E.g., $<$ and $\lesssim$ may be (free existential) predicate variables or λ-terms, which are not terms in first-order languages. As they do not have to interact with our sequents before they are instantiated and applied ($\lambda\beta$-reduced), the language of our deductive logic can still be first-order.

3.2 Semantical Requirements

Validity is expected to be given relative to some Σ-structure (Σ-algebra, Σ-frame) $\mathcal{A}$, assigning a non-empty universe (or "carrier" or "object domain") (to each type).

For $\mathrm{X} \subseteq \mathrm{V}$ we denote the set of total $\mathcal{A}$-valuations of X (i.e. functions mapping variables to objects of the universe of $\mathcal{A}$ (respecting types)) with $\mathrm{X} \to \mathcal{A}$ and the set of (possibly) partial $\mathcal{A}$-valuations of X with $\mathrm{X} \rightsquigarrow \mathcal{A}$.

For $\tau \in \mathrm{X} \to \mathcal{A}$ we denote with '$\mathcal{A}\uplus\tau$' the extension of $\mathcal{A}$ to the variables of X. More precisely, we assume the existence of some evaluation function 'eval' such that $\mathrm{eval}(\mathcal{A}\uplus\tau)$ maps any term whose free occurring symbols are from $\Sigma\uplus\mathrm{X}$ into the universe of $\mathcal{A}$ (respecting types) s.t. for all $x \in \mathrm{X}$:

$$\mathrm{eval}(\mathcal{A}\uplus\tau)(x) = \tau(x)$$

Moreover, $\mathrm{eval}(\mathcal{A}\uplus\tau)$ maps any formula B whose free occurring symbols are from $\Sigma\uplus\mathrm{X}$ to TRUE or FALSE, such that B is valid in $\mathcal{A}\uplus\tau$ iff $\mathrm{eval}(\mathcal{A}\uplus\tau)(B) = \mathsf{TRUE}$.

For any Σ-structure $\mathcal{A}$ with valuation $\tau \in \mathrm{V} \rightsquigarrow \mathcal{A}$, the only additional assumptions we need here are the following two basic properties of most semantics:

Explicitness-Lemma (Andrews (1972), Lemma 2; Andrews (2002), Proposition 5400; Fitting (2002), Proposition 2.30)

The value of the evaluation function on a term or formula B does not depend on the variables that do not occur free in B:
For X being the set of the variables that occur free in B, if $\mathrm{X} \subseteq \mathrm{dom}(\tau)$:

$$\mathrm{eval}(\mathcal{A}\uplus\tau)(B) = \mathrm{eval}(\mathcal{A} \uplus {}_{\mathrm{X}}\upharpoonright\tau)(B).$$

Substitution-Lemma (Andrews (1972), Lemma 3; Andrews (2002), Lemma 5401(a); Enderton (1973), p. 127; Fitting (1996), p. 120; Fitting (2002), Proposition 2.31)

For any substitution σ and term or formula B, if the variables that occur free in $B\sigma$ belong to $\mathrm{dom}(\tau)$:

$$\mathrm{eval}(\mathcal{A}\uplus\tau)(B\sigma) = \mathrm{eval}\big(\ \mathcal{A} \ \uplus\ (\ \sigma\ \uplus\ {}_{\mathrm{V}\setminus\mathrm{dom}(\sigma)}\upharpoonright\mathrm{id}\ )\ \circ\ \mathrm{eval}(\mathcal{A}\uplus\tau)\ \big)\big(\ B\ \big).$$

Note that we take the operator '$\circ$' to have higher priority than the operators '$\cup$' and '$\uplus$'.

Further properties of validity or evaluation are definitely not needed. Note that we have left open what our formulas and what our Σ-structures are. All we need are the above basic requirements. In order not to confuse classical first-order reading, we do not use any special jargon or notation in this paper—except two higher-order ones: Currying, and calling all elements of Σ "constants". Nevertheless, we can assume practically any logic with exactly two truth values here, including intensional, modal,[10] and higher-order[11] logics.

4 Introduction

In this section we introduce the interdisciplinary background required to understand the formal treatment in the following sections. Experts in mathematical induction may esp. skip Section 4.2 and Section 4.4.

4.1 Descente Infinie: A Primitive Example

In this section we present a primitive example proof in the sequent calculus of QuodLibet (cf. Kühler (2000)) in order to give the readers an intuitive understanding of our view of *descente infinie*.

The signature is as follows: We only have the single type nat of natural numbers. We use zero $\mathsf{0} : \mathsf{nat}$ and successor $\mathsf{s} : \mathsf{nat} \to \mathsf{nat}$ as constructors for the type nat. Moreover, $+ : \mathsf{nat} \to \mathsf{nat} \to \mathsf{nat}$ is a defined function on the natural numbers.

We use x, y for variables of the type nat where superscripts like in $x^{\forall,\mathrm{w}}$ indicate a weak free universal variable different from x.

The axioms are as follows:
($\mathsf{nat}1$) $\quad \forall x. \quad \big(x{=}\mathsf{0} \vee \exists y.\ x{=}\mathsf{s}(y) \big)$
(+1) $\quad \forall x. \quad x + \mathsf{0} = x$
(+2) $\quad \forall x, y. \quad x + \mathsf{s}(y) = \mathsf{s}(x + y)$

($\mathsf{nat}1$) says that any natural number is zero or the successor of another natural number, while (+1) and (+2) define the function '+'.

Now we want to prove that $\mathsf{0}$ is neutral also to the left:
(+1sym) $\quad \mathsf{0} + x^{\forall,\mathrm{w}} = x^{\forall,\mathrm{w}};\ \ w_1^{\exists}(x^{\forall,\mathrm{w}}),\ \ <_1^{\exists},\ \ \lesssim_1^{\exists}$

Note that in the syntactical construct (+1sym) the semi-colon is used to separate its sequent from the following weight construct, where the weight term $w_1^{\exists}(x^{\forall,\mathrm{w}})$ measures the induction hypothesis (+1sym) in the induction ordering given by '$<_1^{\exists}$' and '$\lesssim_1^{\exists}$'. Why these explicit weights are so important in *descente infinie* is explained in Wirth & Becker (1995) and more detailed in Wirth (1997), Section 12.

Before we come to our proof tree in XQuodLibet we first give an informal proof in the working mathematician fashion.

We have to show
$$\mathsf{0} + x^{\forall,\mathrm{w}} = x^{\forall,\mathrm{w}}. \tag{1}$$
As, by ($\mathsf{nat}1$), each natural number is either zero or not, we do the following case analysis:

$\underline{x^{\forall,\mathrm{w}} = \mathsf{0}}$: We have to show
$$\mathsf{0} + \mathsf{0} = \mathsf{0}, \tag{1.1}$$
which follows from (+1).

$\underline{x^{\forall,\mathrm{w}} = \mathsf{s}(y^{\forall,\mathrm{w}})}$: We have to show
$$\mathsf{0} + \mathsf{s}(y^{\forall,\mathrm{w}}) = \mathsf{s}(y^{\forall,\mathrm{w}}), \tag{1.2}$$
which by (+2) we can rewrite into
$$\mathsf{s}(\mathsf{0} + y^{\forall,\mathrm{w}}) = \mathsf{s}(y^{\forall,\mathrm{w}}), \tag{1.2.1}$$

which we can again rewrite with the induction hypothesis (1)
(setting $\{x^{\forall,\mathrm{w}} \mapsto y^{\forall,\mathrm{w}}\}$) into

$$\mathsf{s}(y^{\forall,\mathrm{w}}) = \mathsf{s}(y^{\forall,\mathrm{w}}), \qquad \textbf{(1.2.1.1)}$$

which is an equality axiom.

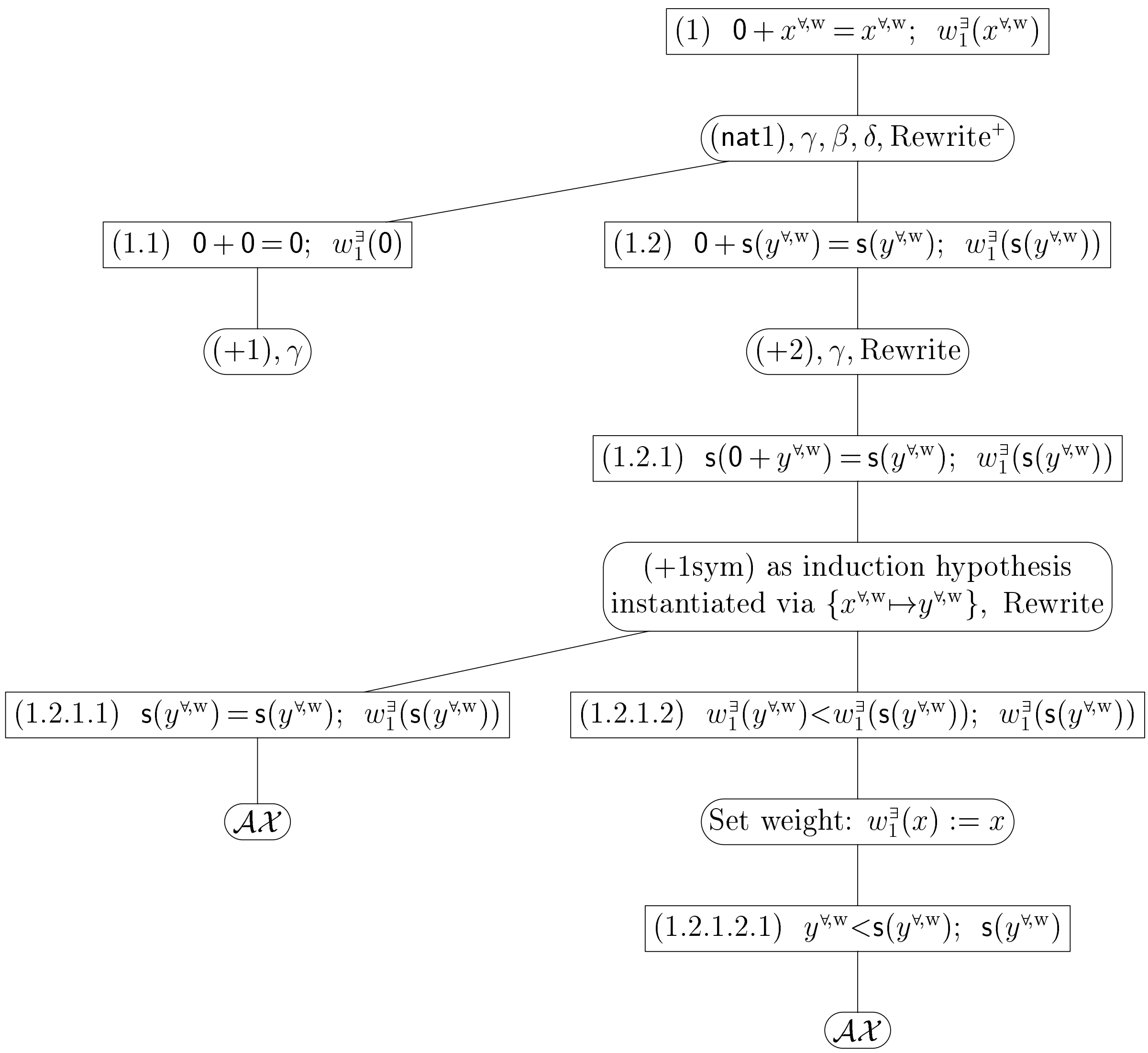
We still have to find an induction ordering $<^{\exists}_{1}$ and some weight term $w^{\exists}_{1}(x^{\forall,\mathrm{w}})$ for " $\mathsf{0} + x^{\forall,\mathrm{w}} = x^{\forall,\mathrm{w}}$ " such that the instance of the applied induction hypothesis is smaller than the induction conclusion we are just proving, i.e. such that $w^{\exists}_{1}(y^{\forall,\mathrm{w}}) <^{\exists}_{1} w^{\exists}_{1}(x^{\forall,\mathrm{w}})$. By our case assumption this is nothing but

$$w^{\exists}_{1}(y^{\forall,\mathrm{w}}) <^{\exists}_{1} w^{\exists}_{1}(\mathsf{s}(y^{\forall,\mathrm{w}})). \qquad \textbf{(1.2.1.2)}$$

But this is trivial: We simply set $w^{\exists}_{1}(x) := x$, choose $<^{\exists}_{1}$ to be the ordering $\prec$ on natural numbers and $\lesssim^{\exists}_{1}$ to be its reflexive ordering $\preceq$, and get the ordering axiom

$$y^{\forall,\mathrm{w}} \prec \mathsf{s}(y^{\forall,\mathrm{w}}). \qquad \textbf{(1.2.1.2.1)}$$

In XQuodLibet the proof tree of this proof will be displayed very similar to the following figure.



Actually, only the square boxes are part of the proof tree. The round-edged boxes show applications of inference rules. While in QuodLibet several inference rules may be applied to a single node of a proof tree (yielding alternative proof attempts in and-or-trees) for the theoretical considerations in this paper we will always apply inference rules only to leaf nodes, resulting in the and-trees that are common for sequent and tableau calculi. Although the round-edged nodes are actually not part of our proof trees here, they are quite useful:

1. In our tree presentation we can check whether the tree is closed simply by realizing that all leaves are round-edged nodes. This means that, in addition to the standard notion of a *tree* (cf. Knuth (1997 f.), Vol. I), as a special feature we assume an explicit representation of leaves, so that, when we add the elements of a set G as children to a leaf node l, this l is not a leaf anymore, even if G is empty. This feature is not only useful for implementation purposes (where we have to record somewhere why a branch is closed) but also in the theory we present in this paper, where the validity of the root sequent of a proof tree is always supposed to reduce to the validity of all leaf sequents of a proof tree and the closed branches should not be considered.

2. We have used round-edged nodes to give some information how inference steps can be achieved in terms of general inference rules like the ones presented at the end of Section 2.3, which are more elementary than the inference rules in QuodLibet.

   E.g., " $(\mathsf{nat1}), \gamma, \beta, \delta$, Rewrite^{+} " in the first round-edged box means that we should actually use the axiom $(\mathsf{nat1})$ and apply a γ-, a β-, and a δ-step and several Rewrite-steps to it in order to get the following partial proof tree below, where in the last inference steps (resulting in (1.1) and (1.2)) the left-most literals of the parents of the leaf nodes are safely (cf. Section 13.2) thrown away (because $x^{\forall,\mathrm{w}}$ is in solved form[12]).

$$(1)$$

$$\neg\forall x.\ (\ x{=}\mathsf{0} \vee \exists y.\ x{=}\mathsf{s}(y)\ ),\ \ \mathsf{0}+x^{\forall,\mathrm{w}}=x^{\forall,\mathrm{w}};\ \ w_1^{\exists}(x^{\forall,\mathrm{w}})$$

$$\neg(\ x^{\forall,\mathrm{w}}{=}\mathsf{0} \vee \exists y.\ x^{\forall,\mathrm{w}}{=}\mathsf{s}(y)\ ),\ \ \mathsf{0}+x^{\forall,\mathrm{w}}=x^{\forall,\mathrm{w}};\ \ w_1^{\exists}(x^{\forall,\mathrm{w}})$$

$$x^{\forall,\mathrm{w}}{\neq}\mathsf{0},\ \ \mathsf{0}+x^{\forall,\mathrm{w}}=x^{\forall,\mathrm{w}};\ \ w_1^{\exists}(x^{\forall,\mathrm{w}}) \qquad \neg\exists y.\ x^{\forall,\mathrm{w}}{=}\mathsf{s}(y),\ \ \mathsf{0}+x^{\forall,\mathrm{w}}=x^{\forall,\mathrm{w}};\ \ w_1^{\exists}(x^{\forall,\mathrm{w}})$$

$$x^{\forall,\mathrm{w}}{\neq}\mathsf{0},\ \ \mathsf{0}+\mathsf{0}=\mathsf{0};\ \ w_1^{\exists}(\mathsf{0}) \qquad x^{\forall,\mathrm{w}}{\neq}\mathsf{s}(y^{\forall,\mathrm{w}}),\ \ \mathsf{0}+x^{\forall,\mathrm{w}}=x^{\forall,\mathrm{w}};\ \ w_1^{\exists}(x^{\forall,\mathrm{w}})$$

$$(1.1) \qquad x^{\forall,\mathrm{w}}{\neq}\mathsf{s}(y^{\forall,\mathrm{w}}),\ \ \mathsf{0}+\mathsf{s}(y^{\forall,\mathrm{w}})=\mathsf{s}(y^{\forall,\mathrm{w}});\ \ w_1^{\exists}(\mathsf{s}(y^{\forall,\mathrm{w}}))$$

$$(1.2)$$

Note that $<_1^\exists$ and $\lesssim_1^\exists$ are not used in the XQuodLibet proof because the system provides only a single induction quasi-ordering $\lesssim$ and no different choice for $\lesssim_1^\exists$ and $<_1^\exists$ but $\lesssim$ and its ordering $<$, resp., is possible, cf. Section 4.2.

4.2 Example: Induction Ordering in QuodLibet

In QuodLibet, a tactic-based ITP system for clausal logic, cf. Kühler (2000), we essentially use the size of a uniquely denoting constructor ground term in the standard ordering on natural numbers for each user-defined type. In each of the models that establish the inductive validity for QuodLibet (type-C in Wirth & Gramlich (1994b)) this results in a wellfounded quasi-ordering on the objects of each universe because different constructor ground terms of the same type evaluate to different objects in the universe for this type.

Moreover, there is a special type ORD that can be used for lexicographic combinations of the user-defined types up to a fixed finite length, say m. Note that we cannot take arbitrary length because the lexicographic combination of arbitrary length of wellfounded orderings is not wellfounded: $(1) > (0,1) > (0,0,1) > \cdots$. This m is not limiting the QuodLibet system, however, because it is not implemented: If a proof attempt is successful it has used only a finite number of finite terms and we can assume that m is the maximum length of lexicographic combination occurring in them.[13]

While for general *descente infinie* not only the weights but also the induction ordering can be chosen for each proof differently, in QuodLibet it has shown to be adequate to use this fixed wellfounded quasi-ordering (depending on the signature Σ) because the lazy substitution of the 2nd order weight variables during the proofs provides sufficient flexibility for the intended application domain of partially defined recursive functions, cf. Kühler & Wirth (1996), Wirth & Gramlich (1994a).

As in the example proof of Section 4.1, each weight for a conjecture Γ initially is a term $w^\exists(y_0^{\forall,\mathrm{w}}, \ldots, y_{n-1}^{\forall,\mathrm{w}})$ where the $y_i^{\forall,\mathrm{w}}$ are the weak free universal variables of Γ and $w^\exists$ is a global (rigid) free existential variable that can be chosen during the induction proof appropriately. Most of the time it is sufficient to let $w^\exists$ be some projection $\forall y_0, \ldots, y_{n-1}.\ w^\exists(y_0, \ldots, y_{n-1}) = y_i$, or more formally, to apply the existential R-substitution $\{w^\exists \mapsto \lambda y_0, \ldots, y_{n-1}.\ y_i\}$. When the goal is $w^\exists(t_1, t_2) < w^\exists(t_2, \mathsf{s}(t_1))$ for natural number terms t_i a good idea might be to choose $w^\exists$ to be the addition on natural numbers, or when in another proof we have the goals $w^\exists(t_1, (t_1 + t_2)) < w^\exists(\mathsf{s}(t_1), t_1)$ and $w^\exists(t_1, t_2) < w^\exists(t_1, \mathsf{s}(t_2))$ a good idea might be to choose $w^\exists$ to be the lexicographic combination of length up to 2. Note that for mutual induction with several conjectures it may be necessary to compare lexical tuples of different length, cf. Section 16.

The fixed induction ordering in systems like Nqthm (cf. Boyer & Moore (1988)) or QuodLibet prevent the users from destroying soundness by providing induction orderings that are not wellfounded. Contrary to this, the ITP system Expander (Padawitz (1998)) permits the free choice of any binary relation but the results come with the proviso that this relation is a wellfounded ordering. As wellfoundedness is a property of 2nd order, only in higher-order theorem provers it becomes possible to have variable induction orderings and express their required wellfoundedness in the theorems, as we will do in Section 19.

4.3 Descente Infinie and Foundedness

We still have to explain what the example proof of Section 4.1 has to do with *descente infinie*. To this end, suppose that there is a counterexample for (1), i.e. some natural number $x^{\forall,\mathrm{w}}$ s.t. $0 + x^{\forall,\mathrm{w}} = x^{\forall,\mathrm{w}}$ is not the case. Since all branches of our proof tree are closed, the counterexample must have left the tree during the proof. With all the standard steps of deductive theorem proving this is impossible because validity of the children always implies validity of the parent sequent. Thus, the counterexample must have gone through (1.2) and (1.2.1) and then jumped out into the applied induction hypothesis. This means that $y^{\forall,\mathrm{w}}$ is some counterexample for (1), too. But (1.2.1.2) shows that this counterexample is smaller than our original one. Thus any counterexample would descend infinitely (*descente infinie*). The above proof tree can be seen as a program for computing, for each given natural number, a purely deductive proof tree when we replace the branch (1.2.1.2) with the recursive call to this program again. Since our induction ordering is wellfounded, the sub-tree (1.2.1.2) guarantees termination. Therefore, we know that after a finite number of recursive calls—although this number of descents may be indefinite (*descente indéfinie*)—the program will end up in the branch of the base case (1.1).

Definition 4.1 (Method of Descente Infinie)
A proposition Γ can be proved by *descente infinie* as follows:

> *Show that for each counterexample of Γ there is another counterexample of Γ that is smaller in a wellfounded ordering!*

Finally note that for the proof in the working mathematician fashion it is not really important whether the working mathematician thinks in terms of counterexamples. What matters is that he can execute the proof method. Indeed, the higher notion of *foundedness* (cf. Definition 1 of Wirth & Becker (1995) and Definition 12.2 of this paper) can be applied in order to think about *descente infinie* without the negative argumentation on counterexamples, but with the positive metaphor of building a supporting frame in a swamp, cf. Section 12.

4.4 Dependent Choice, Wellfoundedness, and Descente Infinie

Although the Axiom of Foundation and even very strong forms of the Axiom of Choice cannot destroy a consistency of set theory (cf. Gödel (1986 ff.), Vol. II), this does not mean that it is generally appropriate to assume the validity of these axioms. The Axiom of Choice is especially inappropriate when one wants to discuss the logical strength of different forms of induction. This is because the Axiom of Choice implies the strongest known forms of induction, so that all forms of induction become valid and of equal logical strength in the presence of the Axiom of Choice. A weak form (or proper logical consequence) of (the strong proper class form of) the Axiom of Choice is the following; cf. Rubin & Rubin (1985), p. 19; Howard & Rubin (1998), Form 43, p. 30.

Definition 4.2 (Principle of Dependent Choice)
If R is a binary relation with $\mathrm{ran}(R) \subseteq \mathrm{dom}(R) \neq \emptyset$, then R is not terminating.

While we have defined wellfoundedness over the existence of minimal elements in classes, a well-known alternative is to define it as termination of the reverse relation. While the converse of the following principle is tautological, the principle itself is not. Thus it makes wellfoundedness independent of the choice of one of its alternatives for its definition; cf. Howard & Rubin (1998), Form 43 R, p. 32.

Definition 4.3 (Principle of Wellfoundedness)
If $<$ is an ordering and $>$ is terminating, then $<$ is wellfounded.

Definition 4.4 (Principle of Descente Infinie)
If $<$ is an ordering, the class A has no $<$-minimal elements (i.e. $\forall a \in A.\ \exists a' \in A.\ a > a'$), and

Version 1: $> \cap\, (A \times A)$ is terminating

Version 2: each $C \subseteq A$ totally ordered by $<$ has a $<$-minimal element

then A is empty.

When we used the alternative definition of wellfoundedness, we would need[14] the Principle of Descente Infinie to guarantee the soundness of the Method of Descente Infinie, cf. Definition 4.1. Indeed, the soundness is achieved by setting A to be the class of counterexamples of Γ in Version 1, which is slightly stronger only at first sight than Version 2, which we got listed in Howard & Rubin (1998), p. 31, as Form 43 K. In fact, all these principles are of equal strength:

Lemma 4.5 *The Principles of Dependent Choice, Wellfoundedness, and Descente Infinie (both versions) are logically equivalent to each other in set theory, even without axioms of Choice, Foundation, or Power-set.*

Finally, note that theoretically it is also possible to use the strictly stronger Axiom of Choice (or Zorn's Lemma) instead of the Principle of Dependent Choice in order to get the soundness principle for a stronger induction method than the Method of Descente Infinie by replacing in Definition 4.4 " $> \cap\, (A \times A)$ is terminating" with "each non-terminating sequence in $> \cap\, (A \times A)$ has a $< \cap\, (A \times A)$ -lower bound", cf. Geser (1995).

4.5 Without Skolemization

Contrary to most first-order deductive frameworks, *Skolemization* is not appropriate for *descente infinie*, for matrix calculi like the ones in Wallen (1990), and some higher-order approaches like Kohlhase (1998). Skolemization has at least three problematic aspects:

1. Skolemization enriches the signature or introduces higher-order variables. Unless special care is taken, this may introduce objects into empty universes, change the notion of term-generatedness or Herbrand or Henkin models, and imply forms of the Axiom of Choice that were not part of the theory before. Above that, the Skolem functions occur in answers to queries or solutions of constraints which in general cannot be translated into the original signature. For a detailed discussion of these problems cf. Miller (1992).

2. Skolemization results in the following simplified quantification structure:

   For all Skolem functions $\vec{u}$ there are solutions to the free existential variables $\vec{e}$ (i.e. the free variables of Fitting (1996)) s.t. the quantifier-free theorem $\Gamma(\vec{e}, \vec{u})$ is valid. Short: $\forall \vec{u}.\ \exists \vec{e}.\ \Gamma(\vec{e}, \vec{u})$

   When the state of a proof attempt is represented as the conjunction of the branches of a tree (as e.g. in sequent or (dually) in tableau calculi), the free existential variables become "rigid" or "global", i.e. a solution for a free existential variable must solve all occurrences of this variable in the whole proof tree. This is because, for $B_0, \ldots, B_n$ denoting the branches of the proof tree for $\Gamma(\vec{e}, \vec{u})$, $\forall \vec{u}.\ \exists \vec{e}.\ (\ B_0 \wedge\ \ldots\ \wedge B_n\ )$ is in general logically strictly stronger than $\forall \vec{u}.\ (\ \exists \vec{e}.\ B_0\ \wedge\ \ldots\ \wedge\ \exists \vec{e}.\ B_n\ )$

   Moreover, with this quantification structure it does not seem to be possible to do ITP by *descente infinie* because the induction hypothesis applications may destroy the counterexample:

   When we have some counterexample $\vec{u}$ for $\Gamma(\vec{e}, \vec{u})$ (i.e. there is no $\vec{e}$ s.t. $\Gamma(\vec{e}, \vec{u})$ is valid) then, for different $\vec{e}$, different branches B_i in the proof tree may cause the invalidity of the conjunction. If we have applied induction hypotheses in more than one branch, for different $\vec{e}$ we get different smaller counterexamples for different branches. What we would need, however, is one single smaller counterexample for all $\vec{e}$.

3. Skolemization increases the size of the formulas. (Note that in most calculi the only relevant part of Skolem terms is the top symbol and the set of occurring variables.)

The first and second problematic aspects disappear when one uses *raising* (cf. Miller (1992)) instead of Skolemization. Raising is a dual of Skolemization and simplifies the quantification structure to something like:

There are raising functions $\vec{e}$ s.t. for all possible values of the free universal variables $\vec{u}$ (i.e. the nullary constants or "parameters") the quantifier-free theorem $\Gamma(\vec{e}, \vec{u})$ is valid. Short: $\exists \vec{e}.\ \forall \vec{u}.\ \Gamma(\vec{e}, \vec{u})$

Note that due to the two duality switches "unsatisfiability vs. validity" and "Skolemization vs. raising", in this paper raising will look much like Skolemization in refutational theorem proving. The difference between raising and Skolemization is best remembered by the fact that raising is related to the γ-rules while Skolemization is related to the δ-rules according to the classification of Section 2.3.

The inverted order of universal and existential quantification of raising (compared to Skolemization) is advantageous because now $\exists \vec{e}.\ \forall \vec{u}.\ (\ B_0 \wedge\ \ldots\ \wedge B_n\ )$ is indeed logically equivalent to $\exists \vec{e}.\ (\ \forall \vec{u}.\ B_0\ \wedge\ \ldots\ \wedge\ \forall \vec{u}.\ B_n\ )$

Furthermore, the induction hypothesis application of *descente infinie* works well:

When, for some (fixed) $\vec{e_0}$, we have some counterexample $\vec{u}$ for $\Gamma(\vec{e_0}, \vec{u})$ (i.e. $\Gamma(\vec{e_0}, \vec{u})$ is invalid) then one branch B_i in the proof tree must cause the invalidity of the conjunction. If this branch is closed, then it contains the application of an induction hypothesis that is invalid for this $\vec{e_0}$ and the $\vec{u}'$ resulting from the instantiation of the hypothesis. Thus, $\vec{u}'$ together with the induction hypothesis provides the strictly smaller counterexample we are looking for for this $\vec{e_0}$.

The third problematic aspect disappears when the dependency of variables is explicitly represented in a *variable-condition*, cf. Kohlhase (1995). This idea actually has a long history, cf. Prawitz (1960), Kanger (1963), Bibel (1987). Moreover, the use of variable-conditions frees our free existential variables from carrying around the free universal variables they may depend on. Thus, the free existential variables for first-order bound variables stay first-order.

4.6 Preservation of Solutions

Users even of pure Prolog are not so much interested in theorem proving as they are in computation of answers to queries or of solutions to query variables. The theorem they want to prove usually contains some free existential variables that are instantiated during a proof attempt. When the proof attempt is successful, not only the input theorem is known to be valid but also the instance of the theorem with the substitution built-up during the proof. Since the knowledge of mere existence is less useful than the knowledge of a term that witnesses this existence, theorem proving should—if possible without overhead—always provide these witnessing terms (or solutions). Computation of answers and solutions is no problem for Prolog's Horn logic theories. For clausal logic theories, in Baumgartner &al. (1997) tableau calculi are used for computation of answers that go beyond simple solutions (i.e. substitutions) in that they are sets of substitutions s.t. the disjunction of the instantiations of the query with these substitutions is in the theory; a possible negation of the query instances is, however, not taken into account. This negation is included in Ten Cate & Shan (2002), where for general first-order theories the answers computed with a tableau calculus are formulas built-up from instances of the query and equalities with any logical operators and quantifiers. For universal theories and quantifier-free queries, these answers do not need quantifiers and are very intuitive. In general, however, when δ-steps occur in a proof, the introduced free universal variables (or Skolem terms) may provide no information on what kind of object they denote. While this is not possible in terms of computability or λ-terms, the information can be provided in form of Hilbert's ε-terms (cf. Leisenring (1969), Hilbert & Bernays (1968/70), Vol. II), and our calculi will offer this possibility. In order to avoid additional overhead, in this paper, however, we will focus on *preservation of solutions* instead of computation of answers, which is a more general task. By "preservation of solutions" we mean at least the following property:

> *All solutions that transform a proof attempt for a proposition into a closed proof (i.e. the closing substitutions for the free existential variables) are also solutions of the original proposition.*

Suppose that our original input theorem $\mathit{\Gamma}(\vec{e}, \vec{u})$ (cf. the discussion in Section 4.5) has been reduced to $G(\vec{e}, \vec{u})$ representing the state of the proof attempt. With "preservation of solutions" we mean that, for any instance $\vec{e_0}$,

$$G(\vec{e_0}, \vec{u}) \text{ implies } \mathit{\Gamma}(\vec{e_0}, \vec{u}) \text{ for each } \vec{u}. \tag{\$}$$

This is again closely related to *descente infinie*:

> Suppose that we have found some instance $\vec{e_0}$ s.t. for each counterexample $\vec{u}$ of $G(\vec{e_0}, \vec{u})$ there is a counterexample $\vec{u}'$ for the original theorem (i.e. $\Gamma(\vec{e_0}, \vec{u}')$ is invalid) and that this $\vec{u}'$ is strictly smaller than $\vec{u}$ in some wellfounded ordering. In this case we have proved $\Gamma(\vec{e_0}, \vec{u})$ (and thus $\Gamma(\vec{e}, \vec{u})$) only if
>
> each counterexample $\vec{u}$ for $\Gamma(\vec{e_0}, \vec{u})$ is also a counterexample for $G(\vec{e_0}, \vec{u})$.
>
> The latter is the contrapositive of ($) and therefore logically equivalent to it.

4.7 The Liberalized δ-rule

Definition 4.6 (Variable-Condition)
A *variable-condition* is a subset of $\mathrm{V}_{\mathrm{free}} \times \mathrm{V}_{\mathrm{free}}$.

Roughly speaking, for a variable-condition R, $(x^{\exists}, y^{\forall}) \in R$ says that $x^{\exists}$ is older than $y^{\forall}$, so that we must not instantiate the free existential variable $x^{\exists}$ with a term containing the free universal variable $y^{\forall}$.

While the benefit of the introduction of free existential variables in γ-rules is to delay the choice of a witnessing term (which is required by our design goal of a natural flow of information, cf. Section 2.1), it is sometimes unsound to instantiate a free existential variable $x^{\exists}$ with a term containing a free universal variable $y^{\forall}$ that was introduced later than $x^{\exists}$:

Example 4.7 $\exists x.\ \forall y.\ (x = y)$
is not deductively valid. We can start a proof attempt via:

γ-step:	$\forall y.\ (x^{\exists} = y),\quad \exists x.\ \forall y.\ (x = y)$
δ-step:	$(x^{\exists} = y^{\forall}),\quad \exists x.\ \forall y.\ (x = y)$

Now, if we were allowed to substitute the free existential variable $x^{\exists}$ with the free universal variable $y^{\forall}$, we would get the tautology $(y^{\forall} = y^{\forall})$, i.e. we would have proved an invalid formula. In order to prevent this, the δ-step has to record $(x^{\exists}, y^{\forall})$ in the variable-condition, which disallows the instantiation step.

In order to restrict the possible instantiations as little as possible, we should keep our variable-conditions as small as possible. Kanger (1963), Bibel (1987), and Wallen (1990) are quite generous in that they let their variable-conditions become quite big:

Example 4.8 $\exists x.\ \big(\ \forall y.\ \neg\mathsf{P}(y)\ \vee\ \mathsf{P}(x)\ \big)$
can be proved the following way:

γ-step:	$\forall y.\ \neg\mathsf{P}(y)\ \vee\ \mathsf{P}(x^{\exists}),\quad \exists x.\ \big(\ \forall y.\ \neg\mathsf{P}(y)\ \vee\ \mathsf{P}(x)\ \big)$
α-step:	$\forall y.\ \neg\mathsf{P}(y),\quad \mathsf{P}(x^{\exists}),\quad \exists x.\ \big(\ \forall y.\ \neg\mathsf{P}(y)\ \vee\ \mathsf{P}(x)\ \big)$
Liberalized δ-step:	$\neg\mathsf{P}(y^{\forall,\mathrm{s}}),\quad \mathsf{P}(x^{\exists}),\quad \exists x.\ \big(\ \forall y.\ \neg\mathsf{P}(y)\ \vee\ \mathsf{P}(x)\ \big)$
Instantiation step:	$\neg\mathsf{P}(y^{\forall,\mathrm{s}}),\quad \mathsf{P}(y^{\forall,\mathrm{s}}),\quad \exists x.\ \big(\ \forall y.\ \neg\mathsf{P}(y)\ \vee\ \mathsf{P}(x)\ \big)$

The last step is not allowed in the above citations, so that another γ-step must be applied to the original formula in order to prove it. Our instantiation step, however, is perfectly sound: Since $x^{\exists}$ does not occur in $\forall y.\ \neg\mathsf{P}(y)$, the free variables $x^{\exists}$ and $y^{\forall,\mathrm{s}}$ do not depend on each other and there is no reason to insist on $x^{\exists}$ being older than $y^{\forall,\mathrm{s}}$. Indeed, moving-in the existential quantifier transforms the original formula into the logically equivalent formula

$\forall y.\ \neg\mathsf{P}(y)\ \vee\ \exists x.\ \mathsf{P}(x)$, which (after a preceding α-step) enables the δ-step introducing $y^{\forall,\mathrm{s}}$ to come before the γ-step introducing $x^{\exists}$.

Keeping the variable-conditions generated by the δ-rule small results in exponential and even non-elementary reduction of the size of smallest proofs. The "liberalization of the δ-rule" has the following history of reduction in size of smallest proofs: Smullyan (1968), Hähnle & Schmitt (1994) (δ^{+}), Beckert &al. (1993) (δ^{++}), Baaz & Fermüller (1995) (δ^{*}), Cantone & Nicolosi-Asmundo (2000) ($\delta^{*^{*}}$). The step from δ^{+} to δ^{++} (just like the step from δ^{++} to Giese & Ahrendt (1999) (δ^{ε})) does not reduce the variable-condition (as all others do) but reduces the number of Skolem symbols (just like the step from δ^{*} to $\delta^{*^{*}}$). While the liberalized δ-rule of Smullyan (1968) is already able to prove the formula of Ex. 4.8 with a single γ-step, it is much more restrictive than the δ^{+}-rule which can treat free existential variables. For this paper, the change from the non-liberalized δ-rule to the liberalized δ^{+}-rule is the problematic one because it destroys the preservation of solutions, cf. Section 8. Some further improvements over δ^{+} are considered in Section 20.

Note that liberalization of the δ-rule is not a simple task because it easily results in unsound calculi, cf. Kohlhase (1995) w.r.t. our Ex. 4.9 and Kohlhase (1998) w.r.t. our Ex. 14.2. The difficulty lies with instantiation steps that relate previously unrelated variables:

Example 4.9 $\exists x.\ \forall y.\ \mathsf{Q}(x,y)\ \vee\ \exists u.\ \forall v.\ \neg\mathsf{Q}(v,u)$ is not deductively valid (to wit, let Q be the identity relation on a non-trivial universe). Consider the following proof attempt: One α-, two γ-, and two liberalized δ-steps result in

$$(*)\qquad \mathsf{Q}(x^{\exists},y^{\forall,\mathrm{s}}),\quad \neg\mathsf{Q}(v^{\forall,\mathrm{s}},u^{\exists}),\quad \exists x.\ \forall y.\ \mathsf{Q}(x,y),\quad \exists u.\ \forall v.\ \neg\mathsf{Q}(v,u)$$

with variable-condition

$$(\#)\qquad R\ :=\ \{(x^{\exists},y^{\forall,\mathrm{s}}),\quad (u^{\exists},v^{\forall,\mathrm{s}})\}$$

Note that the non-liberalized δ-rule would additionally have produced $(x^{\exists},v^{\forall,\mathrm{s}})$ or $(u^{\exists},y^{\forall,\mathrm{s}})$ or both, depending on the order of the proof steps. When we now instantiate $x^{\exists}$ with $v^{\forall,\mathrm{s}}$, we relate the previously unrelated variables $u^{\exists}$ and $y^{\forall,\mathrm{s}}$. Thus, our new goal

$$\mathsf{Q}(v^{\forall,\mathrm{s}},y^{\forall,\mathrm{s}}),\quad \neg\mathsf{Q}(v^{\forall,\mathrm{s}},u^{\exists}),\quad \exists x.\ \forall y.\ \mathsf{Q}(x,y),\quad \exists u.\ \forall v.\ \neg\mathsf{Q}(v,u)$$

must be equipped with the new variable-condition $(u^{\exists},y^{\forall,\mathrm{s}})$. Otherwise we could instantiate $u^{\exists}$ with $y^{\forall,\mathrm{s}}$, resulting in the tautology $\mathsf{Q}(v^{\forall,\mathrm{s}},y^{\forall,\mathrm{s}}),\ \neg\mathsf{Q}(v^{\forall,\mathrm{s}},y^{\forall,\mathrm{s}}),\ \ldots$
Note that in the standard framework of Skolemization and unification, this new variable-condition is automatically generated by the occur-check of unification: When we instantiate $x^{\exists}$ with $v^{\forall,\mathrm{s}}(u^{\exists})$ in $\mathsf{Q}(x^{\exists},y^{\forall,\mathrm{s}}(x^{\exists})),\quad \neg\mathsf{Q}(v^{\forall,\mathrm{s}}(u^{\exists}),u^{\exists}),\quad \ldots$
we get $\mathsf{Q}(v^{\forall,\mathrm{s}}(u^{\exists}),y^{\forall,\mathrm{s}}(v^{\forall,\mathrm{s}}(u^{\exists}))),\quad \neg\mathsf{Q}(v^{\forall,\mathrm{s}}(u^{\exists}),u^{\exists}),\quad \ldots$
which cannot be reduced to a tautology because $y^{\forall,\mathrm{s}}(v^{\forall,\mathrm{s}}(u^{\exists}))$ and $u^{\exists}$ cannot be unified. When we instantiate the variables $x^{\exists}$ and $u^{\exists}$ in the sequence $(*)$ in parallel via

$$(\$)\qquad \sigma := \{x^{\exists}\mapsto v^{\forall,\mathrm{s}},\ u^{\exists}\mapsto y^{\forall,\mathrm{s}}\},$$

we have to check whether the newly imposed variable-conditions are consistent with the substitution itself. In particular, a cycle as given (for the R of $(\#)$) by

$$\begin{tikzcd} u^{\exists} \arrow[r, "R"'] & v^{\forall,\mathrm{s}} \arrow[d, "\sigma^{-1}"'] \\ y^{\forall,\mathrm{s}} \arrow[u, "\sigma^{-1}"'] & x^{\exists} \arrow[l, "R"'] \end{tikzcd}$$

must not exist.

5 Existential Substitutions

Several binary relations on free variables will be introduced in this and the following sections. The overall idea to be remembered is that when (x, y) occurs in such a relation this means something like "x is necessarily older than y" or "the value of y depends on or is described in terms of x".

Definition 5.1 (E_σ, U_σ)
For a substitution σ we define the *existential relation* to be

$$E_\sigma \;:=\; \{\ (z^\exists, x) \mid x \in \mathrm{dom}(\sigma) \wedge z^\exists \in \mathcal{V}_\exists(\sigma(x))\ \},$$

and the *universal relation* to be

$$U_\sigma \;:=\; \{\ (y^\forall, x) \mid x \in \mathrm{dom}(\sigma) \wedge y^\forall \in \mathcal{V}_\forall(\sigma(x))\ \}.$$

Definition 5.2 ([[Quasi-]Existential] R-Substitution)
Let R be a variable-condition, cf. Definition 4.6.
σ is an *R-substitution* if σ is a substitution for that $R \cup E_\sigma \cup U_\sigma$ is wellfounded.
σ is *existential* if $\mathrm{dom}(\sigma) \subseteq \mathrm{V}_\exists$. σ is *quasi-existential* if $\mathrm{dom}(\sigma) \subseteq \mathrm{V}_\exists \uplus \mathrm{V}_{\forall,\mathrm{S}}$.

Note that, regarding syntax, $(x, z^\exists) \in R$ is intended to mean that an R-substitution σ may not replace x with a term in which $z^\exists$ occurs, i.e. $(z^\exists, x) \in E_\sigma$ must be disallowed, i.e. $R \cup E_\sigma$ must be wellfounded. As another example, take from Ex. 4.9 the variable-condition R of (#) and the σ of ($). As explained there, σ must not be an R-substitution due to the cycle

$$\begin{tikzcd} u^\exists \arrow[r, "R"'] & v^{\forall,\mathrm{S}} \arrow[d, "U_\sigma"'] \\ y^{\forall,\mathrm{S}} \arrow[u, "U_\sigma"'] & x^\exists \arrow[l, "R"'] \end{tikzcd}$$

which just contradicts the wellfoundedness of $R \cup U_\sigma$. Note that in practice w.l.o.g. R, E_σ, and U_σ can always be chosen to be finite, so that $R \cup E_\sigma \cup U_\sigma$ is wellfounded iff it is acyclic.

After application of an R-substitution σ, in case of $(x, y^\forall) \in R$, we have to update our variable-condition R in order to ensure that x is not replaced with $y^\forall$ via a future application of another R-substitution that replaces a free variable say $u^\exists$ occurring in $\sigma(x)$ with $y^\forall$. In this case, the transitive closure of the updated variable-condition has to contain $(u^\exists, y^\forall)$. But we have $u^\exists\ E_\sigma\ x\ R\ y^\forall$. This means that $R \cup E_\sigma$ must be a subset of the updated variable-condition. Besides this, we have to add steps with U_σ again.

Definition 5.3 (σ-Update)
Let R be a variable-condition and σ be a substitution. The *σ-update of R* is $R \cup E_\sigma \cup U_\sigma$.

Example 5.4
In the proof attempt of Ex. 4.9 we applied the existential R-substitution $\sigma' = \{x^{\exists} \mapsto v^{\forall,\mathrm{s}}\}$ where $R = \{(x^{\exists}, y^{\forall,\mathrm{s}}),\ (u^{\exists}, v^{\forall,\mathrm{s}})\}$. Note that $U_{\sigma'} = \{(v^{\forall,\mathrm{s}}, x^{\exists})\}$ and $E_{\sigma'} = \emptyset$. Thus, the σ'-update of R is then given by the following finite graph whose transitive closure is irreflexive and thus a wellfounded ordering.

$$\begin{tikzcd} u^{\exists} \arrow[r, "R"'] & v^{\forall,\mathrm{s}} \arrow[d, "U_{\sigma'}"'] \\ y^{\forall,\mathrm{s}} & x^{\exists} \arrow[l, "R"'] \end{tikzcd}$$

Note that our treatment of variable-conditions here is not arbitrary, but carefully designed according to the following items.

- For efficiency reasons we should never compute transitive closures but simply keep adding new edges to a graph. The relevant wellfoundedness-checks can then be performed as acyclicity-checks, which have a time complexity that is linear in the number of edges. This cannot be improved.
- All other approaches I found in the literature are either more complicated and less powerful or else unsound, cf. Section 4.7. The version presented here is optimal for the choices described in the following two items.
- It is with our variable-conditions that we have invested most effort in the possibility of an efficient implementation and where we are already very close to an implementation. Actually, we have even sacrificed some other possibilities for efficiency; namely for the liberalization of the δ-rule (cf. Section 4.7) we sacrificed the possibility to represent Henkin quantifiers or K. Jaakko J. Hintikka's IF logic, cf. Hintikka (1996). Nevertheless, you can have them if you really want them.[15]
- For simplicity of implementation we even have made re-use and permutations of free existential variables like $\{x^{\exists} \mapsto u^{\exists},\ u^{\exists} \mapsto x^{\exists}\}$ impossible. Indeed, all these substitutions lead to a cycle in the existential relation and thus are no R-substitutions according to the above definitions. Re-use and permutations of free existential variables are not appropriate in practice because then we need a time reference in addition to the name of a free existential variable to retrieve its solution or value. Nevertheless, you can have them if you really want them, and we have worked out everything for you in a sequence of notes[16] because this non-trivial enterprise put to test the well-designedness of the concepts of the whole modeling presented in this paper.

6 Existential Valuations

Let $\mathcal{A}$ be some Σ-structure. We now define semantical counterparts of our existential R-substitutions, which we will call "existential $(\mathcal{A}, R)$-valuations".

As such an existential $(\mathcal{A}, R)$-valuation e takes over the role of the raising functions of Section 4.5, it does not simply map each free existential variable directly to an object of $\mathcal{A}$ (of the same type), but must have the ability to additionally read the values of some free universal variables under an $\mathcal{A}$-valuation $\tau \in \mathrm{V}_{\forall} \to \mathcal{A}$. More precisely, e gets some restriction of τ, say $\tau' \in \mathrm{V}_{\forall} \rightsquigarrow \mathcal{A}$ with $\tau' \subseteq \tau$, as a second argument. Short:

$$e : \mathrm{V}_{\exists} \to (\mathrm{V}_{\forall} \rightsquigarrow \mathcal{A}) \rightsquigarrow \mathcal{A}.$$

Moreover, for each free existential variable $x^{\exists}$, we require the set $\mathrm{dom}(\tau')$ of free universal variables read by $e(x^{\exists})$ to be identical for all τ. This identical set will be denoted with $S_e\langle\{x^{\exists}\}\rangle$ below. More technically, we require that there is some "semantical relation" $S_e \subseteq \mathrm{V}_{\forall} \times \mathrm{V}_{\exists}$ s.t. for all $x^{\exists} \in \mathrm{V}_{\exists}$:

$$e(x^{\exists}) \;:\; (S_e\langle\{x^{\exists}\}\rangle \to \mathcal{A}) \to \mathcal{A}.$$

Note that, for each $e : \mathrm{V}_{\exists} \to (\mathrm{V}_{\forall} \rightsquigarrow \mathcal{A}) \rightsquigarrow \mathcal{A}$, at most one semantical relation exists, namely

$$S_e \;:=\; \{\ (y^{\forall}, x^{\exists}) \mid x^{\exists} \in \mathrm{V}_{\exists} \wedge y^{\forall} \in \mathrm{dom}(\textstyle\bigcup(\mathrm{dom}(e(x^{\exists}))))\ \}.$$

In the following definitions we are slightly more general because we want to apply the terminology not only to free existential variables but also to strong free universal variables.

Definition 6.1 (Semantical Relation (S_e))
The *semantical relation of* e is

$$S_e \;:=\; \{\ (y, x) \mid x \in \mathrm{dom}(e) \wedge y \in \mathrm{dom}(\textstyle\bigcup(\mathrm{dom}(e(x))))\ \}.$$

e is *semantical* if e is a partial function on V s.t. for all $x \in \mathrm{dom}(e)$:

$$e(x) : (S_e\langle\{x\}\rangle \to \mathcal{A}) \to \mathcal{A}.$$

Definition 6.2 (Existential $(\mathcal{A}, R)$-Valuation)
Let R be a variable-condition and $\mathcal{A}$ a Σ-structure. e is an *existential $(\mathcal{A}, R)$-valuation* if $e : \mathrm{V}_{\exists} \to (\mathrm{V}_{\forall} \rightsquigarrow \mathcal{A}) \rightsquigarrow \mathcal{A}$, e is semantical, and $R \cup S_e$ is wellfounded.

Finally, we need the technical means ϵ that turns an existential $(\mathcal{A}, R)$-valuation e together with a valuation τ of the free universal variables into a valuation $\epsilon(e)(\tau)$ of the free existential variables:

Definition 6.3 (ϵ)
We define the function

$$\epsilon : \quad (\mathrm{V} \rightsquigarrow (\mathrm{V} \rightsquigarrow \mathcal{A}) \rightsquigarrow \mathcal{A}) \quad \to \quad (\mathrm{V} \rightsquigarrow \mathcal{A}) \quad \to \quad \mathrm{V} \quad \rightsquigarrow \quad \mathcal{A}$$

for $\quad e : \mathrm{V} \rightsquigarrow (\mathrm{V} \rightsquigarrow \mathcal{A}) \rightsquigarrow \mathcal{A}, \quad \tau \in \mathrm{V} \rightsquigarrow \mathcal{A}, \quad x \in \mathrm{V}$

by $\quad \epsilon(e)(\tau)(x) := e(x)({}_{S_e\langle\{x\}\rangle}{\upharpoonright}\tau).$

7 Validity

We are now going to define R-validity of a set of sequents with free variables, in terms of validity of a formula.

Definition 7.1 (Validity, K**)**
Let R be a variable-condition, $\mathcal{A}$ a Σ-structure, and G a set of sequents.
G is *R-valid in $\mathcal{A}$* if there is an existential $(\mathcal{A}, R)$-valuation e s.t. G is $(e, \mathcal{A})$-valid.
G is *$(e, \mathcal{A})$-valid* if G is $(\tau, e, \mathcal{A})$-valid for all $\tau \in \mathrm{V}_\forall \to \mathcal{A}$.
G is *$(\tau, e, \mathcal{A})$-valid* if G is valid in $\mathcal{A} \uplus \epsilon(e)(\tau) \uplus \tau$.
G is *valid in $\mathcal{A}$* if Γ is valid in $\mathcal{A}$ for all $\Gamma \in G$.
A sequent Γ is *valid in $\mathcal{A}$* if there is some formula listed in Γ that is valid in $\mathcal{A}$.
Validity in a class of Σ-structures is understood as validity in each of the Σ-structures of that class.
If we omit the reference to a special Σ-structure we mean validity &c. in some fixed class K of Σ-structures, e.g. the class of all Σ-structures (Σ-algebras) or the class of Herbrand Σ-structures (term-generated Σ-algebras), cf. Wirth & Gramlich (1994b) for more interesting classes for establishing inductive validities.

Example 7.2 (Validity)

For $x^\exists \in \mathrm{V}_\exists$, $y^\forall \in \mathrm{V}_\forall$, the sequent $x^\exists{=}y^\forall$ is $\emptyset$-valid in any $\mathcal{A}$ because we can choose $S_e := \mathrm{V}_\forall{\times}\mathrm{V}_\exists$ and $e(x^\exists)(\tau) := \tau(y^\forall)$ for $\tau \in \mathrm{V}_\forall \to \mathcal{A}$, resulting in $\epsilon(e)(\tau)(x^\exists) = e(x^\exists)({}_{S_e\langle\!\{x^\exists\}\!\rangle}{\upharpoonright}\tau) = e(x^\exists)({}_{\mathrm{V}_\forall}{\upharpoonright}\tau) = \tau(y^\forall)$. This means that $\emptyset$-validity of $x^\exists{=}y^\forall$ is the same as validity of $\forall y.\ \exists x.\ x{=}y$. Moreover, note that $\epsilon(e)(\tau)$ has access to the τ-value of $y^\forall$ just as a raising function f for x in the raised (i.e. dually Skolemized) version $f(y^\forall){=}y^\forall$ of $\forall y.\ \exists x.\ x{=}y$.

Contrary to this, for $R := \mathrm{V}_\exists{\times}\mathrm{V}_\forall$, the same formula $x^\exists{=}y^\forall$ is not R-valid in general because then the required irreflexivity of $S_e{\circ}R$ implies $S_e{=}\emptyset$, and $e(x^\exists)({}_{S_e\langle\!\{x^\exists\}\!\rangle}{\upharpoonright}\tau) = e(x^\exists)({}_{\emptyset}{\upharpoonright}\tau) = e(x^\exists)(\emptyset)$ cannot depend on $\tau(y^\forall)$ anymore. This means that $(\mathrm{V}_\exists{\times}\mathrm{V}_\forall)$-validity of $x^\exists{=}y^\forall$ is the same as validity of $\exists x.\ \forall y.\ x{=}y$. Moreover, note that $\epsilon(e)(\tau)$ has no access to the τ-value of $y^\forall$ just as a raising function c for x in the raised version $c{=}y^\forall$ of $\exists x.\,\forall y.\,x{=}y$.

For a more general example let $G = \{\ A_{i,0} \ldots A_{i,n_i-1} \mid i \in I\ \}$, where for $i \in I$ and $j \prec n_i$ the $A_{i,j}$ are formulas with free existential variables from $\vec{x}$ and free universal variables from $\vec{y}$. Then $(\mathrm{V}_\exists{\times}\mathrm{V}_\forall)$-validity of G means validity of $\exists\vec{x}.\ \forall\vec{y}.\ \forall i \in I.\ \exists j \prec n_i.\ A_{i,j}$
whereas $\emptyset$-validity of G means validity of $\forall\vec{y}.\ \exists\vec{x}.\ \forall i \in I.\ \exists j \prec n_i.\ A_{i,j}$

8 Motivation for Reduction and Strong Validity

Besides the notion of validity we need the notion of reduction. Roughly speaking, a set G_0 of sequents reduces to a set G_1 of sequents if validity of G_1 implies validity of G_0. This, however, is too weak for our purposes here because we are not only interested in validity but also in preserving the solutions for the free existential variables: For ITP, computation of answers and solutions, and constraint solving it is important that the solutions of G_1 are also solutions of G_0. Thus, we could define that G_0 *R-reduces to* G_1 if $(e, \mathcal{A})$-validity of G_1 implies $(e, \mathcal{A})$-validity of G_0 for each existential $(\mathcal{A}, R)$-valuation e. This definition works well with all inference rules at the end of Section 2.3 with the exception of the liberalized δ-rules.

The additional solutions (or existential substitutions) that result from the smaller variable-condition generated by the liberalized δ-rule admit additional proofs and answer computations compared to the (non-liberalized) δ-rule. These additional solutions do not add much difficulty when one is interested in validity only, cf. e.g. Hähnle & Schmitt (1994), but when also the preservation of solutions is required, they pose some problems because they may tear some strong free universal variable, say $y^{\forall,\mathrm{s}}$, out of its context, namely out of the scope of the quantifier eliminated by $y^{\forall,\mathrm{s}}$:

Example 8.1 (Reduction & Liberalized δ-Steps)
In Ex. 4.8 a liberalized δ-step reduced $\quad \forall y.\ \neg\mathsf{P}(y),\quad \mathsf{P}(x^{\exists}),\ \ldots$
to $\quad \neg\mathsf{P}(y^{\forall,\mathrm{s}}),\quad \mathsf{P}(x^{\exists}),\ \ldots$
with empty variable-condition $R := \emptyset$. The latter sequent is $(e, \mathcal{A})$-valid for the existential $(\mathcal{A}, R)$-valuation e given by

$$e(x^{\exists})(\tau) := \tau(y^{\forall,\mathrm{s}}).$$

The former sequent, however, is not $(e, \mathcal{A})$-valid when $\mathsf{P}^{\mathcal{A}}(a)$ is true and $\mathsf{P}^{\mathcal{A}}(b)$ is false for some a, b from the universe of $\mathcal{A}$. To see this, take some τ with $\tau(y^{\forall,\mathrm{s}}) := b$.

How can we solve the problem exhibited in Ex. 8.1? I.e. how can we change the notion of reduction such that the liberalized δ-step becomes a reduction step?

The first approach one may try is to allow a slight modification of e to e' such that $e'(x^{\exists})(\tau) = a$. However, such a modification of e does not go together well with our requirement of preservation of solutions. Besides, this first approach eventually fails because it is not possible to preserve reduction under Instantiation steps:

> E.g., an Instantiation step with the existential R-substitution $\{x^{\exists} \mapsto y^{\forall,\mathrm{s}}\}$ transforms the reduction of Ex. 8.1 into the reduction of $\quad \forall y.\ \neg\mathsf{P}(y),\quad \mathsf{P}(y^{\forall,\mathrm{s}})$
> to $\quad \neg\mathsf{P}(y^{\forall,\mathrm{s}}),\quad \mathsf{P}(y^{\forall,\mathrm{s}})$
>
> Taking τ, e, and $\mathcal{A}$ as in Ex. 8.1, the new latter sequent is still $(e, \mathcal{A})$-valid. There is, however, no modification e' of e such that the new former sequent is $(\tau, e', \mathcal{A})$-valid.

Learning from this, the second approach one may try is to allow a slight modification of τ instead. E.g., for the reduction step of Ex. 8.1, one would require the existence of some $\pi \in \{y^{\forall,\mathrm{S}}\} \to \mathcal{A}$ s.t. the former sequent is $(\pi \uplus {}_{\mathrm{V}_\forall\setminus\{y^{\forall,\mathrm{S}}\}}\!\upharpoonright\!\tau,\ e,\ \mathcal{A})$-valid instead of $(\tau, e, \mathcal{A})$-valid. Choosing $\pi := \{y^{\forall,\mathrm{S}}\mapsto a\}$ would solve the problem of Ex. 8.1 then: Indeed, the former sequent is $(\pi \uplus {}_{\mathrm{V}_\forall\setminus\{y^{\forall,\mathrm{S}}\}}\!\upharpoonright\!\tau,\ e,\ \mathcal{A})$-valid because for the e of Ex. 8.1 we have $e(x^\exists)(\pi \uplus {}_{\mathrm{V}_\forall\setminus\{y^{\forall,\mathrm{S}}\}}\!\upharpoonright\!\tau) = (\pi \uplus {}_{\mathrm{V}_\forall\setminus\{y^{\forall,\mathrm{S}}\}}\!\upharpoonright\!\tau)(y^{\forall,\mathrm{S}}) = a$.

Moreover, with this approach, reduction is preserved under Instantiation steps.

The problems with this approach arise, however, when one asks whether there has to be a single π for all τ or, for each τ, a different π:

Example 8.2
Consider the following liberalized δ-step where the additional free universal variable $z^\forall$ occurs in the principal formula, namely the reduction of $\quad \forall y.\ z^\forall{\neq}y, \quad z^\forall{=}x^\exists$
to $\quad z^\forall{\neq}y^{\forall,\mathrm{S}}, \quad z^\forall{=}x^\exists$

For the e of Ex. 8.1 (which gives $x^\exists$ the value of $y^{\forall,\mathrm{S}}$) the latter sequent is $(e, \mathcal{A})$-valid.

Different π: In case of $R{=}\emptyset$, the former sequent must be $(\pi \uplus {}_{\mathrm{V}_\forall\setminus\{y^{\forall,\mathrm{S}}\}}\!\upharpoonright\!\tau, e, \mathcal{A})$-valid for all τ. This can only hold when the $\pi \in \{y^{\forall,\mathrm{S}}\} \to \mathcal{A}$ can change when the τ-value of $z^\forall$ changes:

E.g., for $\tau := \{y^{\forall,\mathrm{S}}\mapsto a,\ \ z^\forall\mapsto b\}$ we need $\pi(y^{\forall,\mathrm{S}}) := b$,
while for $\tau := \{y^{\forall,\mathrm{S}}\mapsto b,\ \ z^\forall\mapsto a\}$ we need $\pi(y^{\forall,\mathrm{S}}) := a$.

Indeed, in the reduction above, $y^{\forall,\mathrm{S}}$ is functionally dependent on $z^\forall$. This dependency is the main[17] reason for our requirement on our liberalized δ-rule to insert $(z^\forall, y^{\forall,\mathrm{S}})$ into the variable-condition, cf. the end of Section 2.3.

Single π: In case of $R{=}\{(x^\exists, z^\forall)\}$, the former sequent of Ex. 8.2 is not R-valid in general. Thus, in order to preserve the connection between reduction and validity (cf. Lemma 11.2(1)), the step of Ex. 8.2 must not be a reduction, i.e. the former sequent must not be $(\pi \uplus {}_{\mathrm{V}_\forall\setminus\{y^{\forall,\mathrm{S}}\}}\!\upharpoonright\!\tau, e, \mathcal{A})$-valid for all τ. Therefore, π must not depend on the τ-value of $z^\forall$, contrary to the item above. Note that such a dependency would effectively allow $x^\exists$ to read the value of $z^\forall$, which is explicitly forbidden by the variable-condition R.

Thus, the only solution can be that π (just like e) depends on some values of τ but not on others. Since we are interested in extracting information on the solution of free existential variables of the original theorem from a completed proof, we want to have the additional possibility to look up what role the strong free universal variables introduced by liberalized δ-steps really play. And this is what the *choice-conditions* of the following section are about. With the help of these choice-conditions we can then define the notion of *strong validity* which gives the strong free universal variables an extra treatment and solves the problem of Ex. 8.1 by disallowing the value b for $y^{\forall,\mathrm{S}}$ via a choice-condition that requires to choose a value for $y^{\forall,\mathrm{S}}$ such that $\mathsf{P}(y^{\forall,\mathrm{S}})$ becomes true.

9 Choice-Conditions

As will be explained below, choice-conditions are closely related to Hilbert's ε-terms, cf. Hilbert & Bernays (1968/70), Vol. II and Leisenring (1969). For a motivational introduction to choice-conditions as an indefinite semantics for Hilbert's ε-terms, cf. Wirth (2002). Note that the optional part $[\cdots]$ in the following definition is only needed for modeling "subordinate" ε-terms (or "untergeordnete" ε-Ausdrücke according to Hilbert & Bernays (1968/70), Vol. II, p. 24) and not for the combination of *descente infinie* and deduction where we only need ε-binding on top level.

Definition 9.1 (Choice-Condition, Extension)
C is an *R-choice-condition* if R is a wellfounded variable-condition, C is a partial function from $\mathrm{V}_{\forall,\mathrm{s}}$ into the set of formulas, and $z\ R^{+}\ y^{\forall,\mathrm{s}}$ for all $y^{\forall,\mathrm{s}} \in \mathrm{dom}(C)$ and $z \in \mathcal{V}_{\mathrm{free}}(C(y^{\forall,\mathrm{s}}))\backslash\{y^{\forall,\mathrm{s}}\}$.

$\Big[$ More generally, the values of C can be formula-valued λ-terms where,
for $y^{\forall,\mathrm{s}} \in \mathrm{dom}(C)$ and $C(y^{\forall,\mathrm{s}}) = \lambda v_0.\ \ldots \lambda v_{l-1}.\ B$,
B is a formula whose free occurring variables from $\mathrm{V}_{\mathrm{bound}}$
are among $\{v_0, \ldots, v_{l-1}\} \subseteq \mathrm{V}_{\mathrm{bound}}$
and where, for $v_0 : \alpha_0, \ldots, v_{l-1} : \alpha_{l-1}$, we have
$y^{\forall,\mathrm{s}} : \alpha_0 \to \ldots \to \alpha_{l-1} \to \alpha_l$ for some type α_l,
and any occurrence of $y^{\forall,\mathrm{s}}$ in B must be of the form $y^{\forall,\mathrm{s}}(v_0)\cdots(v_{l-1})$. $\Big]$

(C', R') is an *extension of* (C, R) if C is an R-choice-condition, C' is an R'-choice-condition, $C{\subseteq}C'$, and $R{\subseteq}R'$.

Definition 9.2 (Compatibility)
Let C be an R-choice-condition, $\mathcal{A}$ a Σ-structure, and e an existential $(\mathcal{A}, R)$-valuation. π is $(e, \mathcal{A})$*-compatible with* (C, R) if

1. $\pi : \mathrm{V}_{\forall,\mathrm{s}} \to (\mathrm{V}_{\forall,\mathrm{w}} \rightsquigarrow \mathcal{A}) \rightsquigarrow \mathcal{A}$ is semantical (cf. Definition 6.1) and $R \cup S_e \cup S_\pi$ is wellfounded.

2. For all $y^{\forall,\mathrm{s}} \in \mathrm{dom}(C)$ with $C(y^{\forall,\mathrm{s}}) = \lambda v_0.\ \ldots \lambda v_{l-1}.\ B$ and $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$ and $\chi \in \{v_0, \ldots, v_{l-1}\} \to \mathcal{A}$:

   If, for some $\eta \in \{y^{\forall,\mathrm{s}}\} \to \mathcal{A}$,
   B is $\big(\ {}_{\mathrm{V}_{\forall,\mathrm{s}}\backslash\{y^{\forall,\mathrm{s}}\}}\!\upharpoonright\!(\epsilon(\pi)(\tau)) \uplus \eta \uplus \tau \uplus \chi,\ \ e,\ \ \mathcal{A}\ \big)$-valid,
   then B is $(\epsilon(\pi)(\tau) \uplus \tau \uplus \chi, e, \mathcal{A})$-valid.

Item 1 of this definition is quite technical and needed for lemma application. Roughly speaking, it says that the flow of information between variables expressed in R, e, and π is acyclic.

Item 2 of $(e, \mathcal{A})$-compatibility of π with say $\big(\ \{(y^{\forall,\mathrm{s}},\ \lambda v_0.\ \ldots \lambda v_{l-1}.\ B)\},\ \ R\ \big)$ means that a different choice for the $\epsilon(\pi)(\tau)$-value of $y^{\forall,\mathrm{s}}$ cannot give rise to a previously not given validity of the formula B in $\mathcal{A} \uplus \epsilon(\ e\ )(\ \epsilon(\pi)(\tau) \uplus \tau\ ) \uplus \epsilon(\pi)(\tau){\uplus}\tau{\uplus}\chi$, or that $\epsilon(\pi)(\tau)(y^{\forall,\mathrm{s}})$ is chosen such that B becomes valid if such a choice is possible. This is closely related to Hilbert's ε-operator in the sense that $y^{\forall,\mathrm{s}}$ is given the value of

$$\lambda v_0.\ \ldots \lambda v_{l-1}.\ \varepsilon y.\ (B\{y^{\forall,\mathrm{s}}(v_0)\cdots(v_{l-1}) \mapsto y\})$$

for an arbitrary $y \in \mathrm{V}_{\mathrm{bound}}\backslash\mathcal{V}(B)$.

As the choice for $y^{\forall,s}$ depends on the other free variables of $\lambda v_0.\ \ldots\lambda v_{l-1}.\,B$ (i.e. the free variables of $\lambda v_0.\ \ldots\lambda v_{l-1}.\ \varepsilon y.\ (B\{y^{\forall,s}(v_0)\cdots(v_{l-1})\mapsto y\})$), we required the inclusion of this dependency into the transitive closure of the variable-condition R in Definition 9.1.

Note that the empty function $\emptyset$ is an R-choice-condition for any wellfounded (which in the following will always be the case) variable-condition R. Moreover, any $\pi : \mathrm{V}_{\forall,s} \to \{\emptyset\} \to \mathcal{A}$ is $(e,\mathcal{A})$-compatible with $(\emptyset, R)$ due to $S_\pi = \emptyset$. Indeed, a compatible π always exists:

Lemma 9.3
Let C be an R-choice-condition, $\mathcal{A}$ a Σ-structure, and e an existential $(\mathcal{A}, R)$-valuation. Now, there is some π that is $(e,\mathcal{A})$-compatible with (C, R).

Finally, we need means for expressing the requirement on a quasi-existential substitution to replace the strong free universal variables in a way that goes together well with the compatibility of Definition 9.2(2):

Definition 9.4 ($Q_{C,\sigma}$)
For a substitution σ and an R-choice-condition C, we require $Q_{C,\sigma}$ to be a function from $\mathrm{dom}(C) \cap \mathrm{dom}(\sigma)$ into the set of sequents s.t. for each $y^{\forall,s} \in \mathrm{dom}(C) \cap \mathrm{dom}(\sigma)$ with $C(y^{\forall,s}) = \lambda v_0.\ \ldots\lambda v_{l-1}.\ B$, we have $Q_{C,\sigma}(y^{\forall,s}) =$

$$\forall v_0.\ \ldots\forall v_{l-1}.\ (\ \exists y.\ (B\{y^{\forall,s}(v_0)\cdots(v_{l-1})\mapsto y\}) \ \Rightarrow\ B\ )\sigma$$

for an arbitrary $y \in \mathrm{V}_{\mathrm{bound}} \setminus \mathcal{V}(C(y^{\forall,s}))$.

After global application of an R-substitution σ we now have to update both R and C:

Definition 9.5 (Extended σ-Update)
Let C be an R-choice-condition and σ a substitution.
The *extended σ-update* (C', R') *of* (C, R) is given by:

$$C' := \{\ (x, B\sigma) \mid (x,B) \in C \wedge x \notin \mathrm{dom}(\sigma)\ \},$$

R' is the σ-update of R.

Lemma 9.6 (Extended σ-Update)
Let C be an R-choice-condition, σ an R-substitution, and (C', R') the extended σ-update of (C, R). Now: C' is an R'-choice-condition.

10 Strong Validity

Definition 10.1 (Strong Validity)
Let C be an R-choice-condition, $\mathcal{A}$ a Σ-structure, and G a set of sequents.
G is *(C,R)-strongly valid in $\mathcal{A}$* if there is an existential $(\mathcal{A},R)$-valuation e s.t. G is (C,R)-strongly $(e,\mathcal{A})$-valid.
G is *(C,R)-strongly $(e,\mathcal{A})$-valid* if for some[18] π that is $(e,\mathcal{A})$-compatible with (C,R), G is strongly $(\pi,e,\mathcal{A})$-valid.
G is *strongly $(\pi,e,\mathcal{A})$-valid* if G is $(\epsilon(\pi)(\tau)\uplus\tau,e,\mathcal{A})$-valid for each $\tau\in\mathrm{V}_{\forall,\mathrm{w}}\to\mathcal{A}$.
The rest is given by Definition 7.1.

Note that strong validity is called "strong" because it treats the strong free universal variables properly, whereas (weak) validity of Definition 7.1 does not. It is generally not the case, however, that strong validity is logically stronger than weak validity. The logical strength of the two cannot be compared easily, but we do not need to know more than the following two lemmas.

Lemma 10.2 (From Weak to Strong Validity)
Let C be an R-choice-condition, $\mathcal{A}$ a Σ-structure, and G a set of sequents. Now:
If G is $\mathrm{V}_{\exists}\times\mathrm{V}_{\forall}$-valid in $\mathcal{A}$, then G is R-valid and (C,R)-strongly valid in $\mathcal{A}$.

On the other hand, from (C,R)-strong validity of a set of sequents G we should be able to infer $(\emptyset,R')$-strong validity and (weak) R'-validity for some G with the strong free universal variables renamed to some new free existential variables.

Lemma 10.3 (From Strong to Weak Validity)
Let C be an R-choice-condition, $\mathcal{A}$ a Σ-structure, and G a set of sequents.
Let $\varsigma\in\mathcal{V}_{\forall,\mathrm{s}}(G)\to(\mathrm{V}_{\exists}\setminus\mathcal{V}(G))$ be injective.
If G is (C,R)-strongly valid in $\mathcal{A}$, then $G\varsigma$ is $(\emptyset,R')$-strongly valid and (weakly) R'-valid in $\mathcal{A}$ for any $R' \subseteq ({}_{\mathrm{V}_{\forall,\mathrm{w}}\cup\mathrm{V}_{\exists}\setminus\mathrm{ran}(\varsigma)}{\upharpoonleft}\mathrm{id}\uplus\varsigma^{-1})\circ R^{+}{\upharpoonright}_{\mathrm{V}_{\forall,\mathrm{w}}\cup\mathrm{V}_{\exists}\setminus\mathrm{ran}(\varsigma)}\uplus\mathrm{V}_{\exists}\times\mathrm{V}_{\forall,\mathrm{s}}$.

11 Reduction

Definition 11.1 (Reduction)
Let C be an R-choice-condition, $\mathcal{A}$ a Σ-structure, and G_0, G_1 sets of sequents.
G_0 *(C,R)-reduces to G_1 in $\mathcal{A}$* if for each existential $(\mathcal{A},R)$-valuation e and each π that is $(e,\mathcal{A})$-compatible with (C,R):

if G_1 is strongly $(\pi,e,\mathcal{A})$-valid, then G_0 is strongly $(\pi,e,\mathcal{A})$-valid.

In the following lemma skip the very technical optional part $[\cdots]$ on a first reading! (If you nevertheless want to understand the meaning of O and N, have a look at the text preceding Lemma A.2.)

Lemma 11.2 (Reduction)
Let C be an R-choice-condition; $\mathcal{A}$ a Σ-structure; G_0, G_1, G_2, and G_3 sets of sequents.

1. **(Validity)**
Assume that G_0 (C,R)-reduces to G_1 in $\mathcal{A}$. Now: If G_1 is (C,R)-strongly valid in $\mathcal{A}$, then G_0 is (C,R)-strongly valid in $\mathcal{A}$.

2. **(Reflexivity)**
In case of $G_0 \subseteq G_1$: G_0 (C,R)-reduces to G_1 in $\mathcal{A}$.

3. **(Transitivity)**
If G_0 (C,R)-reduces to G_1 in $\mathcal{A}$ and G_1 (C,R)-reduces to G_2 in $\mathcal{A}$, then G_0 (C,R)-reduces to G_2 in $\mathcal{A}$.

4. **(Additivity)**
If G_0 (C,R)-reduces to G_2 in $\mathcal{A}$ and G_1 (C,R)-reduces to G_3 in $\mathcal{A}$, then $G_0 \cup G_1$ (C,R)-reduces to $G_2 \cup G_3$ in $\mathcal{A}$.

5. **(Monotonicity)**
For (C',R') being an extension of (C,R):
   (a) *If G_0 is (C',R')-strongly valid in $\mathcal{A}$, then G_0 is (C,R)-strongly valid in $\mathcal{A}$.*
   (b) *If G_0 (C,R)-reduces to G_1 in $\mathcal{A}$, then G_0 (C',R')-reduces to G_1 in $\mathcal{A}$.*

6. **(Instantiation)**
For an [quasi-] existential R-substitution σ and the extended σ-update (C',R') of (C,R) [and for O, N with $O \subseteq \mathrm{dom}(C) \cap \mathrm{dom}(\sigma) \subseteq O \uplus N$,
$$N \subseteq \mathrm{dom}(C) \setminus O, \quad \mathrm{dom}(C) \cap \langle N \rangle R^{+} \subseteq N, \quad \textit{and} \quad N \cap \mathcal{V}(G_0, G_1) = \emptyset \,]:$$
   (a) *If $G_0\sigma\,[\cup\,\langle O \rangle Q_{C,\sigma}]$ is (C',R')-strongly valid in $\mathcal{A}$, then G_0 is (C,R)-strongly valid in $\mathcal{A}$.*
   (b) *If G_0 (C,R)-reduces to G_1 in $\mathcal{A}$, then $G_0\sigma$ (C',R')-reduces to $G_1\sigma\,[\cup\,\langle O \rangle Q_{C,\sigma}]$ in $\mathcal{A}$.*

12 Counterexamples and Foundedness

For powerful ITP we have to be able to restrict the test of whether the weight of a hypothesis is smaller than the weight of a goal (which has to be satisfied for the permission to apply the hypothesis to the goal) to the special case semantically described by their logic parts. This can be achieved by considering only such instances of their weights that result from valuations that describe invalid instances of their logic parts. A syntactical construct (cf. Definition 3.2) augmented with such a valuation providing extra information on the invalidity of its logic part in some Σ-structure $\mathcal{A}$ is our formal means to capture the notion of "counterexample".

Definition 12.1 (Counterexample)
Let $\mathcal{A}$ be a Σ-structure from K, C be an R-choice-condition, e be an existential $(\mathcal{A}, R)$-valuation and π be $(e, \mathcal{A})$-compatible with (C, R).
(S, τ) is an $(\pi, e, \mathcal{A})$-*counterexample (for S)* if S is a syntactical construct, $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$, and $\mathrm{logic}(\{S\})$ is not $(\epsilon(\pi)(\tau) \uplus \tau, e, \mathcal{A})$-valid. (Thus, the logic part of a syntactical construct S is strongly $(\pi, e, \mathcal{A})$-valid iff S has no $(\pi, e, \mathcal{A})$-counterexamples.)

Now we come to the crucial notion notion of *foundedness*, which for ITP is what the notion of reduction is for deduction. Due to its technical complexity, before coming to its definition, let us motivate the notion of foundedness with the metaphor of building a supporting frame in a swamp.

1. We can fix a construction element H to a construction element H' on a strictly lower level of the supporting frame resulting in the construction

   $$\begin{array}{c} H \\ \downarrow \\ H' \end{array}$$

   for which we also write $H \searrow H'$ and which expresses that if H needs some support, then it can get it from the element H' below.

   In the world of induction this means that if an element of the set H has a counterexample, then there is a counterexample for an element of H' whose weight term is strictly smaller in the induction ordering $<$ that must be identical for both counterexamples.

2. We can fix a construction element H to a construction element (G, L) on the same or lower level of the supporting frame resulting in the construction

   $$H \curvearrowright (G, L)$$

   In the world of induction this means that if an element of H has a counterexample, then there is a counterexample for an element of G or L. Moreover, if this counterexample is from G, then it has to be smaller or equal in the induction quasi-ordering $\lesssim$ that must be identical for both counterexamples. Note that 'H' stands for the induction $\underline{\text{h}}$ypotheses, 'G' for the sub-$\underline{\text{g}}$oals of H, and 'L' for the $\underline{\text{l}}$emmas of the proof.

3. We can fix a construction element H partly to a construction element (G, L) on the same or lower level and partly to a construction element H' on a strictly lower level of the supporting frame resulting in the construction

$$\begin{array}{l} H \curvearrowright (G, L) \\ \downarrow \\ H' \end{array}$$

for which we also write $H \searrow\!/\!\curvearrowright (H', G, L)$. In the world of induction this means that if an element of H has a counterexample, then there is a counterexample for an element of H', G, or L. Moreover, if this counterexample is from H' then it has to be strictly smaller and if it is from G it has to be equal or smaller than the original counterexample from H in the induction ordering they share.

Now, if we have a supporting frame of the form $H \searrow\!/\!\curvearrowright (H, G, L)$, i.e.

$$\begin{array}{l} H \curvearrowright (G, L) \\ \downarrow \\ H \curvearrowright (G, L) \\ \downarrow \\ \vdots \end{array}$$

and we know that the swamp is wellfounded (i.e. we get to solid ground everywhere if we only go deep enough) then we know that H is sufficiently supported against sinking by the element (G, L) alone, i.e. $H \curvearrowright (G, L)$. In the world of induction this means that all sequents of the elements of the set H are inductively valid provided that the base cases in G and the lemmas in L are, cf. Lemma 12.3(7).

Finally, note that the expressive power of $\searrow\!/\!\curvearrowright$ is higher than that of $\searrow$ and $\curvearrowright$ together: $\{S\} \searrow H \;\vee\; \{S\} \curvearrowright (G, L)$ implies $\{S\} \searrow\!/\!\curvearrowright (H, G, L)$ for $S \in \text{SynCons}$, but the converse does not hold in general because different counterexamples for S may have smaller counterexamples in different sets. Thus we define $\searrow\!/\!\curvearrowright$ first:

Definition 12.2 (Foundedness)
Let C be an R-choice-condition. Let $G_0, G_1, H, L \subseteq \mathrm{SynCons}$. Now:
G_0 is (C,R)-*strict/quasi-founded* on (H, G_1, L) (denoted by $G_0 \searrow\!/\!\curvearrowright_{C,R} (H, G_1, L)$) if $\forall \mathcal{A} \in \mathrm{K}.\ \forall e$ existential $(\mathcal{A}, R)$-valuation. $\forall \pi$ $(e, \mathcal{A})$-compatible with (C, R).

$$\forall S \in G_0.\ \forall \tau.\ \left(\begin{array}{l}
((S,\tau) \text{ is an } (\pi, e, \mathcal{A})\text{-counterexample})\\
\Rightarrow\ \exists S', \tau'.\ \left(\begin{array}{l}
((S',\tau') \text{ is an } (\pi, e, \mathcal{A})\text{-counterexample})\\
\wedge\ \left(\begin{array}{l}
\exists \mathit{\Gamma}, \mathit{\Gamma}', w, w', <, <', \lesssim, \lesssim', \delta, \delta'\bar{w}, \bar{w}', \lhd, \underset{\sim}{\unlhd}.\\
\left(\begin{array}{ll}
 & (\mathit{\Gamma}, (w, <, \lesssim)) = S\\
\wedge & (\mathit{\Gamma}', (w', <', \lesssim')) = S'\\
\wedge & \delta = \epsilon(\pi)(\tau) \uplus \tau\\
\wedge & \delta' = \epsilon(\pi)(\tau') \uplus \tau'\\
\wedge & \bar{w} = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(w)\\
\wedge & \bar{w}' = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta') \uplus \delta')(w')\\
\wedge & \lhd = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(<)\\
\wedge & \lhd = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta') \uplus \delta')(<')\\
\wedge & \underset{\sim}{\unlhd} = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(\lesssim)\\
\wedge & \underset{\sim}{\unlhd} = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta') \uplus \delta')(\lesssim')\\
\wedge & \left(\begin{array}{ll}
 & \left(\begin{array}{ll}
 & S' \in H\\
\wedge & \bar{w}' \lhd^{+} \bar{w}\\
\wedge & \lhd \circ \underset{\sim}{\unlhd} \subseteq \lhd^{+}\\
\wedge & \lhd \text{ is wellfounded}
\end{array}\right)\\
\vee & \left(\begin{array}{ll}
 & S' \in G_1\\
\wedge & \bar{w}'\ (\underset{\sim}{\unlhd} \cup \lhd)^{*}\ \bar{w}
\end{array}\right)
\end{array}\right)
\end{array}\right)\\
\vee\ \ S' \in L
\end{array}\right)
\end{array}\right)
\end{array}\right)$$

G_0 is *strictly* (C,R)-*founded on* H (denoted by $G_0 \searrow_{C,R} H$) if $G_0 \searrow\!/\!\curvearrowright_{C,R} (H, \emptyset, \emptyset)$.
G_0 is (C,R)-*founded on* (G_1, L) (denoted by $G_0 \curvearrowright_{C,R} (G_1, L)$) if $G_0 \searrow\!/\!\curvearrowright_{C,R} (\emptyset, G_1, L)$.

Note that $H \curvearrowright_{C,R} (\emptyset, L)$ iff $\mathrm{logic}(H)$ (C,R)-reduces to $\mathrm{logic}(L)$ in all $\mathcal{A} \in \mathrm{K}$.

Moreover, note that in case of "$<$" &c. being no proper terms of our (possibly first-order) logic language, "$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(<)$" is to be taken a shorthand for

$$\{\ (a,b)\ \mid\ \mathrm{eval}(\ \mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta \uplus \{x \mapsto a,\ y \mapsto b\}\ )(\ x < y\ ) = \mathsf{TRUE}\ \},$$

for two distinct variables $x, y \in \mathrm{V}_{\mathrm{bound}} \setminus \mathcal{V}(<)$.

Furthermore, note that the definition could be simplified by requiring $\underset{\sim}{\unlhd}$ to be a wellfounded quasi-ordering and $\lhd$ to be its ordering. For proof-technical convenience, however, we prefer the weaker requirements: E.g., if we want to formally prove that wellfoundedness of a relation R implies termination of the transitive closure of its reverse, the above conditions should be satisfied when we set $\lhd$ to R (and $\underset{\sim}{\unlhd}$ to $\emptyset$).

Moreover, note that our induction ordering is semantical (cf. Definition 13.7 of Wirth (1997)) in the sense that it cannot depend on the syntactical term structure of a weight w but only on the value of w under the evaluation function. In Wirth (1997) we have rigorously investigated the price one has to pay for the possibility to have induction orderings also depending on the syntax of weights. For powerful concrete inference systems this price turned out to be surprisingly high. Besides this, after improving the ordering information in *descente infinie* by our introduction of explicit weights (cf. Wirth & Becker (1995)) the former necessity of sophisticated induction orderings that exploit the term structure (cf. e.g. Bachmair (1988)) does not seem to exist anymore.

Finally, note that the whole Section 13 depends only on those properties of foundedness that are given in the following lemma (and on the definition of $\curvearrowright_{C,R}$ in terms of $\searrow\!/\!\!\curvearrowright_{C,R}$).

Lemma 12.3 (Foundedness)
Let C be an R-choice-condition; $G_i, G'_i, H_i, L_i \subseteq \mathrm{SynCons}$.

1. **(Validity)**
   Assume $G_0 \curvearrowright_{C,R} (G_1, L_1)$.
   (a) *If* $\mathrm{logic}(G_1 \cup L_1)$ *is (C,R)-strongly valid,*
   then $\mathrm{logic}(G_0)$ *is (C,R)-strongly valid, too.*
   (b) *Let* $\mathcal{A} \in \mathrm{K}$, *$e$ be an existential $(\mathcal{A},R)$-valuation and*
   π be $(e,\mathcal{A})$-compatible with (C,R).
   If $\mathrm{logic}(G_1 \cup L_1)$ *is strongly $(\pi,e,\mathcal{A})$-valid,*
   then $\mathrm{logic}(G_0)$ *is strongly $(\pi,e,\mathcal{A})$-valid, too.*

2. **(Reflexivity)**
   In case of $G_0 \subseteq G_1 \cup L_1$: $\quad G_0 \searrow\!/\!\!\curvearrowright_{C,R} (H_1, G_1, L_1)$.

3. **(Transitivity)**
   (a) *If* $G_0 \curvearrowright_{C,R} (G_1, L_1)$ *and* $G_1 \searrow\!/\!\!\curvearrowright_{C,R} (H_2, G_2, L_2)$,
   then $G_0 \searrow\!/\!\!\curvearrowright_{C,R} (H_2, G_2, L_1 \cup L_2)$.
   (b) *If* $G_0 \curvearrowright_{C,R} (G_1, L_1)$ *and* $L_1 \curvearrowright_{C,R} (G_2, L_2)$, *then* $G_0 \curvearrowright_{C,R} (G_1, G_2 \cup L_2)$.

4. **(Additivity)**
   If $\forall i \in I.\ G_i \searrow\!/\!\!\curvearrowright_{C,R} (H_i, G'_i, L_i)$, *then* $\bigcup_{i\in I} G_i \searrow\!/\!\!\curvearrowright_{C,R} (\bigcup_{i\in I} H_i, \bigcup_{i\in I} G'_i, \bigcup_{i\in I} L_i)$.

5. **(Monotonicity)**
   For (C', R') *being an extension of* (C,R): *If* $G_0 \curvearrowright_{C,R} (G_1, L_1)$, *then* $G_0 \curvearrowright_{C',R'} (G_1, L_1)$.

6. **(Instantiation)**
   For an [quasi-] existential R-substitution σ and
   the extended σ-update (C', R') of (C,R)
   [and for O, N with $O \subseteq \mathrm{dom}(C) \cap \mathrm{dom}(\sigma) \subseteq O \uplus N$,
   $N \subseteq \mathrm{dom}(C) \setminus O$, $\quad \mathrm{dom}(C) \cap \langle N \rangle R^{+} \subseteq N$, *and* $N \cap \mathcal{V}(G_0, G_1, L_1) = \emptyset$,
   and L_2 a set of syntactical constructs with $\mathrm{logic}(L_2) = \langle O \rangle Q_{C,\sigma}$ *]:*

   If $G_0 \curvearrowright_{C,R} (G_1, L_1)$, *then* $G_0\sigma \curvearrowright_{C',R'} (G_1\sigma, L_1\sigma\,[\cup\ L_2])$.

7. **(Descente Infinie)**
   If $H_1 \searrow\!/\!\!\curvearrowright_{C,R} (H_1, G_1, L_1)$, *then* $H_1 \curvearrowright_{C,R} (G_1, L_1)$. [19]

13 Abstract Sequent and Tableau Calculus

Now we are going to abstractly describe inductive sequent and tableau calculi. We will later show that the usual deductive calculi are instances of our abstract calculi and that *descente infinie* can be applied. The benefit of the abstract version is that every instance is automatically sound. Due to the small number of inference rules and the locality of soundness, this abstract version is not really necessary for deductive calculi. For inductive calculi, however, due to a bigger number of inference rules (which usually have to be improved now and then) and the globality of soundness, such an abstract version is very helpful, cf. Wirth & Becker (1995), Wirth (1997). Note that the design of our abstract calculi is not ad hoc. Cf. Wirth & Becker (1995), Wirth (1997) for the discussion of alternatives.

Compared to the well-known deductive proof trees, the difference is that the sequents are replaced with syntactical constructs, i.e. to each sequent a weight construct is added for controlling the loops in ITP. Moreover, in inductive tableau calculi, the proof trees differ from those in deductive tableau calculi in that their roots are labeled with weight constructs instead of formulas.

Definition 13.1 (Proof Forest)
An *(inductive) proof forest in a sequent* (or else: *tableau*) *calculus* is a quintuple

$$(F, C, R, L, H)$$

where C is an R-choice-condition, $L, H \subseteq \mathbf{N}_+ \times \mathbf{N}_+$, and F is a partial function from $\mathbf{N}_+$ into the set of pairs (S, t), where S is a syntactical construct and t is a tree whose nodes are labeled with syntactical constructs (or else: whose root is labeled with a weight construct and whose other nodes are labeled with formulas).

Note that L records the lemma applications and H the induction hypothesis applications as explained below Definition 13.2. Furthermore, the tree t is intended to represent a proof attempt for the hypothesis S. In case of a tableau calculus, the nodes of t are labeled with formulas; the root, however, with a weight construct. In case of a sequent calculus, the nodes are labeled with syntactical constructs. While the syntactical constructs at the nodes of a tree in a sequent calculus stand for themselves as goals, in a tableau calculus all the ancestors have to be included to make up a syntactical construct and, moreover, the labeling formulas are in negated form:

Definition 13.2 (Goals()**, $\mathcal{AX}$, Closedness)**
'Goals(T)' denotes the set of syntactical constructs labeling the leaves of the trees in the set T (or else: the set of syntactical constructs $(\Delta, \sqsupset)$ with Δ resulting from listing the conjugates of the formulas labeling a branch from a leaf to the root (exclusively) in a tree t in T and $\sqsupset$ being the label of the root of the tree t).
In what follows, we assume $\mathcal{AX}$ to be some set of *axioms*. By this we mean that $\mathcal{AX}$ is $\mathrm{V}_\exists \times \mathrm{V}_\forall$-valid (for all $\mathcal{A} \in \mathrm{K}$). (By Lemma 10.2, this means that $\mathcal{AX}$ is R-valid and (R, C)-strongly valid for any R-choice-condition C. For K cf. the last sentence in Definition 7.1.)
Typically, $\mathcal{AX}$ contains all sequents of the forms $\Gamma\, A\, \Pi\, \overline{A}\, \Lambda$ and $\Gamma\,(s{=}s)\,\Pi$ for sequents Γ, Π, Λ, formulas A, and terms s.
The tree t is *closed* if $\mathrm{logic}(\mathrm{Goals}(\{t\})) \subseteq \mathcal{AX}$.

Why do we consider a proof forest instead of a single proof tree only? The possibility to have an empty proof forest provides a nicer starting point. Besides that, if we have two proof trees $F(i) = ((\Gamma, \aleph), t)$ and $F(i') = ((\Gamma', \aleph'), t')$, we can apply Γ' as a lemma in the tree t of $(\Gamma, \aleph)$ and record this lemma application by inserting (i', i) into L. We can also apply $(\Gamma', \aleph')$ instantiated with μ as an induction hypothesis in the tree t of $(\Gamma, \aleph)$ and record this induction hypothesis application by inserting (i', i) into H. In the latter case we have to add additional goals to t that express that the weight term of $\aleph'\mu$ is smaller than the weight term of $\aleph$ and that the induction (quasi-) orderings of $\aleph'\mu$ and $\aleph$ are identical. When only a single constant induction (quasi-) ordering is given (as in QuodLibet, e.g.), the latter equality is trivial and the weight constructs can be simplified to consist of weight terms only. Provided that the lemma application relation $L{\circ}H^*$ is wellfounded and all trees $(i'', (S'', t''))$ with $i''(L{\cup}H)^*i$ are closed, this proves that Γ is (C, R)-valid. Contrary to deductive theorem proving, for *descente infinie* the availability of lemma and induction hypothesis application of this form is really necessary.

The following definition provides some syntactical sugar to make the following more readable.

Definition 13.3 (Hyps(), Trees()**)**
For A being a set of pairs (S, t) consisting of a syntactical construct S and a tree t, we define the hypotheses of A by $\mathrm{Hyps}(A) := \mathrm{dom}(A)$ and the trees of A by $\mathrm{Trees}(A) := \mathrm{ran}(A)$.

In the following definition, abstract proof steps of the following three different kinds are considered: A *Hypothesizing* step starts a new proof tree for a new conjecture; an *Expansion* step expands a proof tree; and an *Instantiation* step globally instantiates some free variables in the whole proof forest. As the definition is quite long and hard, we give some hints directly following the definitions of Expansion and Instantiation.

Definition 13.4 (Abstract Sequent (or else: Tableau) Calculus)

We start with the empty proof forest $(F, C, R, L, H) := (\emptyset, \emptyset, \emptyset, \emptyset, \emptyset)$ and then iterate only the following kinds of modifications of (F, C, R, L, H) (resulting in (F', C', R', L', H')):

Hypothesizing:

Let (C', R') be an extension of (C, R). Set $L' := L$ and $H' := H$. Let $i \in \mathbf{N}_+ \setminus \mathrm{dom}(F)$. Let $(\Gamma, \aleph)$ be a syntactical construct. Let t be the tree with a single node only, which is labeled with $(\Gamma, \aleph)$ (or else: with a single branch only, s.t. Γ is the list of the conjugates of the formulas labeling the branch from the leaf to the root (exclusively) and $\aleph$ is the label of the root).
Then we may set $F' := F \cup \{(i, ((\Gamma, \aleph), t))\}$.

Expansion:

Let (C', R') be an extension of (C, R). Let $N_{\mathrm{L}}, N_{\mathrm{H}} \subseteq \mathrm{dom}(F)$. Set $L' := L \cup N_{\mathrm{L}} \times \{i\}$ and $H' := H \cup N_{\mathrm{H}} \times \{i\}$. Let $(i, (S, t)) \in F$. Let l be a leaf in t. Let $(\Delta, \sqsupset)$ be the label of l (or else: result from listing the conjugates of the formulas labeling the branch from l to the root (exclusively) and $\sqsupset$ be the label of the root of t). Let G be a set of syntactical constructs (or else: let M be a set of sequents and set $G := \{\ (\Pi\Delta, \sqsupset) \mid \Pi \in M\ \}$).

Now if $$\{(\Delta, \sqsupset)\} \searrow\!/\!\curvearrowright_{C', R'} \big(\ \mathrm{Hyps}(\langle N_{\mathrm{H}} \rangle F),\ \ G,\ \ \mathrm{Hyps}(\langle N_{\mathrm{L}} \rangle F)\ \big), \qquad (\$)$$
then we may set $F' := (F \backslash \{(i, (S, t))\}) \cup \{(i, (S, t'))\}$, where t' results from t by adding to the former leaf l, exactly for each syntactical construct S' in G, a new child node labeled with S' (or else: exactly for each sequent Π in M a new child branch s.t. Π is the list of the conjugates of the formulas labeling the branch from the leaf to the new child node of l).

Hints: Expansion steps are parameterized with a syntactical construct $(\Delta, \sqsupset)$, two sets $N_{\mathrm{H}}, N_{\mathrm{L}} \subseteq \mathrm{dom}(F)$, and a set of sequents G s.t. ($) holds. While $\mathrm{Hyps}(\langle N_{\mathrm{H}} \rangle F)$ and $\mathrm{Hyps}(\langle N_{\mathrm{L}} \rangle F)$ contain the hypotheses of the proof trees that are applied as induction hypotheses or lemmas, resp., the syntactical constructs in G become the new child nodes of the former leaf node labeled with $(\Delta, \sqsupset)$. For tableau calculi, however, this set G of syntactical constructs must actually have the form $\{\ (\Pi\Delta, \sqsupset) \mid \Pi \in M\ \}$ because an Expansion step cannot remove formulas from ancestor nodes. This is because these formulas are also part of the goals associated with other leaves in the proof tree.

Instantiation:

Let σ be an [quasi-] existential R-substitution and (C', R') the extended σ-update of (C, R). Set $F' := \{\ (i, ((\Gamma\sigma, \aleph\sigma), t\sigma)) \mid (i, ((\Gamma, \aleph), t)) \in F\ \}$.

[Assume that for each $y^{\forall,\mathrm{s}} \in \mathrm{dom}(C) \cap \mathrm{dom}(\sigma)$ there is some $j_{y^{\forall,\mathrm{s}}} \in \mathrm{dom}(F)$ with
$$\mathrm{logic}(\mathrm{Hyps}(\langle\!\{ j_{y^{\forall,\mathrm{s}}} \}\!\rangle F')) = \{Q_{C,\sigma}(y^{\forall,\mathrm{s}})\}\ .$$
For each $i \in \mathrm{dom}(F)$ set (I abbreviating $H^*\langle\!\{i\}\!\rangle$)
$$D(i) := \mathrm{dom}(C) \cap \mathrm{dom}(\sigma) \cap R^*\Big\langle \mathcal{V}_{\forall,\mathrm{s}}\Big(\mathrm{Goals}(\mathrm{Trees}(\langle I \rangle F)),\ \ \mathrm{Hyps}(\big\langle \{i\} \cup L\langle I \rangle \big\rangle F)\Big)\Big\rangle$$
]

Then we may set $L' := L\ [\ \cup\ \{\ (j_{y^{\forall,\mathrm{s}}}, i) \mid i \in \mathrm{dom}(F) \wedge y^{\forall,\mathrm{s}} \in D(i)\ \}]$ and $H' := H$.

Hints: An Instantiation step globally instantiates free existential and free universal variables via a substitution σ. It is simple unless strong free universal variables are substituted and the parts in optional brackets become relevant. In this case, every replacement of a strong free universal variable $y^{\forall,\mathrm{s}}$ must be justified by a lemma $Q_{C,\sigma}(y^{\forall,\mathrm{s}})$ (cf. Definition 9.4) given by the hypothesis of a proof tree number $j_{y^{\forall,\mathrm{s}}}$, which must be added in a preceding Hypothesizing step unless it is already present. Note that this lemma is special in the sense that it is not applied locally in some proof tree but globally. Esp. problematic is the possibility that $y^{\forall,\mathrm{s}}$ occurs in the proof of the lemma itself. If we are not very careful, the lemma becomes a lemma of itself, resulting in a cyclic lemma application relation. Then our whole proof work is in vain because no validities whatsoever can be inferred. Therefore, since we do not want to reintroduce the lemma as an open lemma and prove it again, we take a very close look on which of our (possibly open) lemmas really depend on the justifying lemma $Q_{C,\sigma}(y^{\forall,\mathrm{s}})$ after global application of σ. Our solution is that the lemma is relevant for any proof tree number i whose proof state (i.e. the goals, the hypothesis itself, and the lemmas, cf. Definition 13.5) contains free variables that may depend on $y^{\forall,\mathrm{s}}$; i.e. for any i with $y^{\forall,\mathrm{s}} \in D(i)$.

13.1 Soundness

The following invariant condition captures the soundness of our proof trees. Roughly speaking (when neither a lemma nor an induction hypothesis has been applied), the validity of the goals of a tree is to imply the validity of the sequent of this tree; short: "The leaves imply the root."

Definition 13.5 (Soundness Invariant Condition)
The *soundness invariant condition on* (F, C, R, L, H) is that (F, C, R, L, H) is a proof forest and that, for all $(i, (S, t)) \in F$,

$$\{S\} \curvearrowright_{C,R} \big(\ \mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F)),\quad \mathrm{Hyps}(\langle L\langle I\rangle\rangle F)\ \big) \ \text{ for } I := H^{*}\langle\!\{i\}\!\rangle.$$

I are the numbers of the proof trees whose hypotheses have been applied in the tree t as induction hypotheses, and $\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F))$ is the set of goals of these proof trees. $\mathrm{Hyps}(\langle L\langle I\rangle\rangle F) = \{\ S' \mid F(i') = (S', t') \wedge i' L i \wedge i \in I\ \}$ are the lemmas on that Γ depends.

Theorem 13.6 (Successful Proof)
Let the proof forest (F, C, R, L, H) *satisfy the above soundness invariant condition.*
Let $(i, ((\Gamma, \aleph), t)) \in F$.
Assume that all trees in $\mathrm{Trees}(\langle (L{\cup}H)^{*}\langle\!\{i\}\!\rangle\rangle F)$
(i.e. in $\{\ t' \mid (i', (S', t')) \in F \ \wedge\ i' \,(L{\cup}H)^{*}\, i\ \}$*)*
are closed and that $L{\circ}H^{*}$ *is wellfounded. Now:*
Γ *is* (C, R)*-strongly valid and, for any injective* $\varsigma : \mathcal{V}_{\forall,\mathrm{s}}(\Gamma) \to (\mathrm{V}_{\exists} \setminus \mathcal{V}(\Gamma))$,
$\Gamma\varsigma$ *is* $(\emptyset, R')$*-strongly valid and (weakly)* R'*-valid for any*
$R' \quad\subseteq\quad \big({}_{\mathrm{V}_{\forall,\mathrm{w}} \cup \mathrm{V}_{\exists} \setminus \mathrm{ran}(\varsigma)}\!\upharpoonleft\!\mathrm{id} \ \uplus\ \varsigma^{-1}\big) \circ R^{+}\!\upharpoonright_{\mathrm{V}_{\forall,\mathrm{w}} \cup \mathrm{V}_{\exists} \setminus \mathrm{ran}(\varsigma)} \quad\uplus\quad \mathrm{V}_{\exists} \times \mathrm{V}_{\forall,\mathrm{s}}.$

Theorem 13.7 (Soundness)
The above soundness invariant condition is always satisfied for the sequent (or else: tableau) calculus of Definition 13.4.

13.2 Safeness

While the soundness invariant condition of Definition 13.5 ("the leaves imply the root") is an essential one, its converse, namely "the root implies the leaves", which we call *safeness*, is useful in practice for failure detection.

Definition 13.8 (Safeness Invariant Condition)
The *safeness invariant condition on* (F, C, R, L, H) is that, for all $(i, ((\Gamma, \aleph), t)) \in F$,

$$\mathrm{logic}(\mathrm{Goals}(\{t\})) \quad (C, R)\text{-reduces to} \quad \{\Gamma\}.$$

Definition 13.9 (Safeness of Steps and Sub-rules)
An Expansion step of the sequent calculus of Definition 13.4 is *safe* if (referring to the variables introduced there) $\mathrm{logic}(G)$ (C', R')-reduces to $\{\Delta\}$.
A sub-rule of the Expansion rule of the sequent calculus of Definition 13.4 is *safe* if it describes only safe Expansion steps.
An Instantiation step of the sequent (or else: tableau) calculus of Definition 13.4 is *safe* if $Q_{C,\sigma}(y^{\forall,\mathrm{s}})$ is (C', R')-strongly valid due to Theor. 13.6 for all $y^{\forall,\mathrm{s}} \in \mathrm{dom}(C) \cap \mathrm{dom}(\sigma)$, i.e. all trees in $\mathrm{Trees}(\langle (L' \cup H')^* \langle\{j_{y^{\forall,\mathrm{s}}}\}\rangle\rangle F')$ are closed and $L' \circ H'^*$ is wellfounded.

No notions of safeness are given for Expansion steps of the tableau calculus and for Hypothesizing steps. This is because these steps preserve the safeness invariant condition trivially. Furthermore, note that an Instantiation step can only be unsafe if strong free universal variables are instantiated.

Theorem 13.10 (Safeness)
The above safeness invariant condition is always satisfied for the tableau calculus of Definition 13.4 if all Instantiation steps are safe.
The above safeness invariant condition is always satisfied for the sequent calculus of Definition 13.4 if all Expansion and Instantiation steps are safe.

Suppose we have disproved a goal of a tree t with $(i, ((\Gamma, \aleph), t)) \in F$, i.e. found out that the goal is invalid. If safeness is provided for the whole construction of t, then we know that Γ is invalid. If some steps in t may be unsafe, we should backtrack up to an "unsafe" step that may have caused this invalidity. When we suppose that all Expansion and Instantiation steps are safe, the only reason not to remove the hypothesis Γ can be that it was valid originally but in the meanwhile invalidated by an instantiation:

- If there have been no Instantiation steps affecting the sequent Γ, then we should remove $(i, ((\Gamma, \aleph), t))$ from the proof forest F and undo all its applications as a lemma or as an induction hypothesis, i.e. e.g. the Expansion steps of Definition 13.4 where i occurs in the sets N_{L}, N_{H}, resp..
- Otherwise, we should undo an Instantiation step affecting the sequent Π, and then see whether we can still detect a failure by again disproving the disinstantiated goal.

14 Concrete Sequent and Tableau Calculus

The examples of α-, β-, γ-, δ-, liberalized δ-, Rewrite-, and Cut-rules at the end of Section 2.3 can be modeled as safe Expansion steps as follows: Let

$$\frac{\Delta}{\Pi_0 \quad \ldots \quad \Pi_{n-1}} \quad \begin{matrix} C'' \\ R'' \end{matrix}$$

denote a sub-rule of the Expansion rule of the sequent calculus of Definition 13.4 which is given by $N_{\mathrm{L}} := N_{\mathrm{H}} := \emptyset$ (i.e. neither lemmas nor induction hypotheses are applied), $G := \{(\Pi_0, \sqsupset), \ldots, (\Pi_{n-1}, \sqsupset)\}$, $C' := C \cup C''$, and $R' := R \cup R''$. For $C'' = R'' = \emptyset$, the 'C''' and 'R''' to the right are omitted.

For such a rule being a safe sub-rule of the Expansion rule of the sequent calculus of Definition 13.4 we have to show that C' is an R'-choice-condition, that $\{(\Delta, \sqsupset)\} \searrow/\curvearrowright_{C',R'} (\emptyset, G, \emptyset)$, and that logic$(G)$ (C', R')-reduces to $\{\Delta\}$.

Theorem 14.1 *Taking $\mathcal{F} = (F, C, R, L, H)$, the examples of α-, β-, γ-, δ-,*[20] *liberalized δ-, Rewrite-, and Cut-rules at the end of Section 2.3 are safe sub-rules of the Expansion rule of the sequent calculus of Definition 13.4.*

Note that the resp. rules for the tableau calculus of Definition 13.4 differ only in M consisting of the sub-sequents containing the new (i.e. the first one or two) formulas of the sequents below the bar.

The following example shows that R'' of the liberalized δ-rule at the end of Section 2.3 must indeed contain $\mathcal{V}_{\forall}(A) \times \{x^{\forall,\mathrm{s}}\}$ and that the transitive closure over R' must be considered for the property of being an existential R'-substitution.

Example 14.2 $\quad \exists y.\, \forall x.\, \big(\, \forall z.\, \mathsf{Q}(x,z) \;\vee\; \neg\mathsf{Q}(x,y) \,\big)$

is not deductively valid (to wit, let Q be the identity relation on a non-trivial universe).

γ-step: $\quad \forall x.\, \big(\, \forall z.\, \mathsf{Q}(x,z) \;\vee\; \neg\mathsf{Q}(x,y^{\exists}) \,\big), \quad \exists y.\, \forall x.\, \big(\, \forall z.\, \mathsf{Q}(x,z) \;\vee\; \neg\mathsf{Q}(x,y) \,\big)$

Liberalized or non-liberalized δ-step:

$$\big(\, \forall z.\, \mathsf{Q}(x^{\forall},z) \;\vee\; \neg\mathsf{Q}(x^{\forall},y^{\exists}) \,\big), \quad \exists y.\, \forall x.\, \big(\, \forall z.\, \mathsf{Q}(x,z) \;\vee\; \neg\mathsf{Q}(x,y) \,\big)$$

with variable-condition $(y^{\exists}, x^{\forall})$.

α-step: $\quad \forall z.\, \mathsf{Q}(x^{\forall},z), \quad \neg\mathsf{Q}(x^{\forall},y^{\exists}), \quad \exists y.\, \forall x.\, \big(\, \forall z.\, \mathsf{Q}(x,z) \;\vee\; \neg\mathsf{Q}(x,y) \,\big)$

Liberalized δ-step: $\quad \mathsf{Q}(x^{\forall},z^{\forall,\mathrm{s}}), \quad \neg\mathsf{Q}(x^{\forall},y^{\exists}), \quad \exists y.\, \forall x.\, \big(\, \forall z.\, \mathsf{Q}(x,z) \;\vee\; \neg\mathsf{Q}(x,y) \,\big)$

with additional choice-condition $(z^{\forall,\mathrm{s}}, \neg\mathsf{Q}(x^{\forall}, z^{\forall,\mathrm{s}}))$ and additional variable-condition $(x^{\forall}, z^{\forall,\mathrm{s}})$, i.e. the current variable-condition R' is given by

$$y^{\exists} \xrightarrow[R]{} x^{\forall} \xrightarrow[R'']{} z^{\forall,\mathrm{s}}$$

Note that now we have $y^{\exists} R'^{+} z^{\forall,\mathrm{s}}$ although $y^{\exists}$ does not appear in $\mathsf{Q}(x^{\forall}, z)$.

Thus, both the inclusion of the free universal variables of the principle formula into the domain of the variable-condition and the transitive closure over it are necessary for $\sigma := \{y^{\exists} \mapsto z^{\forall,\mathrm{s}}\}$ not being an existential R'-substitution in our state of proof. The latter fact is, however, essential for soundness, because without it we could complete the proof attempt by application of σ in an Instantiation step, leading to the tautology

$$\mathsf{Q}(x^{\forall},z^{\forall,\mathrm{s}}), \quad \neg\mathsf{Q}(x^{\forall},z^{\forall,\mathrm{s}}), \quad \exists y.\, \forall x.\, \big(\, \forall z.\, \mathsf{Q}(x,z) \;\vee\; \neg\mathsf{Q}(x,y) \,\big)$$

Now we will present one primitive rule for applying $(\Phi, \daleth)$ instantiated with a substitution ϱ as a lemma or as an induction hypothesis to expand a goal $(\Delta, \beth)$ of a proof tree t. We will formulate this as an Expansion step in the tableau calculus of Definition 13.4 (sequent calculus analogously) as follows.

Theorem 14.3

The following describes two sub-rules of the Expansion rule of the tableau calculus of Definition 13.4:

Let (F, C, R, L, H), $(\Delta, \beth)$, and (C', R') be given as in the Expansion rule.
Let $(j, ((\Phi, \daleth), t'')) \in F$.
Let $Y \subseteq \left\{ \; y^{\forall,\mathrm{w}} \in \mathcal{V}_{\forall,\mathrm{w}}(\Phi, \daleth) \;\middle|\; (\; \mathcal{V}_{\exists}(\Phi, \daleth) \cup \mathcal{V}_{\forall,\mathrm{s}}(\Phi, \daleth) \;) \times \{y^{\forall,\mathrm{w}}\} \;\subseteq\; R' \;\right\}$.
Let $\varrho \in Y \to (\mathrm{V}_{\exists} \setminus \mathcal{V}(F, C, R))$ be an injective substitution.

Lemma Application: *Set $N_{\mathrm{L}} := \{j\}$ and $N_{\mathrm{H}} := \emptyset$. Set M to be the set containing the single-formula sequents $\overline{B\varrho}$ for each formula B listed in the sequent Φ.*

Induction Hypothesis Application: *Set $(w, <, \lesssim) := \beth$ and $(w', <', \lesssim') := \daleth$. Let α be the type of w and w'. Set $N_{\mathrm{L}} := \emptyset$ and $N_{\mathrm{H}} := \{j\}$. Set M to be the set containing the following sequents:*

1. $\overline{B\varrho}$ *for each formula B listed in the sequent Φ*
2. $w'\varrho < w$
3. $\forall p : \alpha \to \mathsf{bool}.\; \left(\exists a : \alpha.\; p(a) \Rightarrow \exists a : \alpha. \left(\begin{array}{c} p(a) \\ \wedge\;\; \neg\exists a' : \alpha. \left(\begin{array}{c} p(a') \\ \wedge\;\; a'{<}a \end{array} \right) \end{array} \right) \right)$

   where "$p(a)$" abbreviates "$p(a){=}\mathsf{true}$" &c. and "$\alpha \to \mathsf{bool}$" must have the standard interpretation of a predicate over "α".
4. $\forall x, y : \alpha.\; \big(\; x < y \;\Leftrightarrow\; x \;({<'}\varrho)\; y \;\big)$
5. $\forall x, y : \alpha.\; \big(\; x \lesssim y \;\Leftrightarrow\; x \;({\lesssim'}\varrho)\; y \;\big)$
6. $\forall x, y, z : \alpha.\; \big(\; (x < y \wedge y \lesssim z) \Rightarrow\; x < z \;\big)$

Each of the sequents (3)-(6) can be omitted if the following holds, resp., for any $\mathcal{A} \in \mathrm{K}$, existential $(\mathcal{A}, R')$-valuation e, and π and τ s.t. π is $(e, \mathcal{A})$-compatible with (C', R') and $((\Delta, \beth), \tau)$ is an $(\pi, e, \mathcal{A})$-counterexample, and for $\delta := \epsilon(\pi)(\tau) \uplus \tau$, $\lhd := \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(<)$, and $\underset{\sim}{\lhd} := \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(\lesssim)$:

3. $\lhd$ *is wellfounded.*
4. $\lhd = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)({<'}\varrho)$.
5. $\underset{\sim}{\lhd} = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)({\lesssim'}\varrho)$.
6. $\lhd \circ \underset{\sim}{\lhd} \subseteq \lhd^{+}$.

Note that Y contains exactly those weak free universal variables of $(\varPhi, \urcorner)$ that have neither free existential variables nor strong free universal variables of $(\varPhi, \urcorner)$ in their scopes when imagining any list of quantifiers for all free variables of $(\varPhi, \urcorner)$ that represents (a superset of) R'. In other words, the variables in Y are those weak free universal variables on that neither a solution for the free existential variables nor a choice-condition for the strong free universal variables in $(\varPhi, \urcorner)$ depends. Therefore, the variables in Y are those which we can instantiate when applying the lemma or induction hypothesis $(\varPhi, \urcorner)$. Although it does not seem impossible to instantiate more variables, this does not seem to be necessary:

- We can extend R' with $(\ \mathcal{V}_{\exists}(\varPhi, \urcorner) \cup \mathcal{V}_{\forall,\mathrm{s}}(\varPhi, \urcorner)\ ) \times \{y^{\forall,\mathrm{w}}\}$ in order to instantiate $y^{\forall,\mathrm{w}}$ when applying the lemma, provided that R' is still wellfounded. If this extension of R' makes a query variable useless (i.e. blocks a solution for a free existential variable), we have to take a higher-order query variable instead, cf. Section 18.

- I do not known any more general approach in the literature. E.g., in Baaz &al. (1997), the inductive part of theorem proving is triggered by application of a δ-rule and the variable y of the quantifier removed by the δ-rule becomes the induction variable. In our approach, the δ-rule application would replace y with a new free universal variable $y^{\forall,\mathrm{w}}$ and extend the variable-condition with $(\mathcal{V}_{\exists}(\varPhi, \urcorner) \cup \mathcal{V}_{\forall,\mathrm{s}}(\varPhi, \urcorner)) \times \{y^{\forall,\mathrm{w}}\}$ so that $y^{\forall,\mathrm{w}} \in \mathrm{Y}$ would hold.

Moreover, note that there is no analogon of Theor. 14.3 using a set of *strong* free universal variables instead of the set Y of *weak* free universal variables. This requires the presence of weak free universal variables even if we are not interested in (non-liberalized) δ-steps. One should always use *weak* free universal variables in Hypothesizing steps. In order to have more useful lemmas and induction hypothesis, we sometimes even have to split a tree at an inner position with a Hypothesizing step introducing a new hypothesis with *weak* free universal variables replacing the *strong* free universal variables and apply this new hypothesis as a lemma to the new leaf of the old tree, closing this branch, cf. the discussion at the end of Section 17.

Furthermore, note that, although Theor. 14.3 does not forbid, it would be silly to destroy the wellfoundedness of $L{\circ}H^*$ required in Theor. 13.6. Thus, it is reasonable to forbid $i\ (L{\cup}H)^*\ j$ for a lemma application and $i\ H^*{\circ}(L{\circ}H^*)^+\ j$ for an induction hypothesis application.

All of the sequents (3)-(6) can be omitted if we (as in QuodLibet, cf. Section 4.2) require that Σ contains (for each type) the binary predicate (or boolean function) symbols $\lesssim$ and $<$ and that each Σ-structure $\mathcal{A}$ assigns to them a wellfounded quasi-ordering $\lesssim^{\mathcal{A}}$ and its ordering $<^{\mathcal{A}}$, resp.. The sequents (5) and (6) can also be omitted in the important special case (cf. Section 19) that the third component $\lessapprox$ of the weight constructs is restricted to be the empty relation $\emptyset$.

Detailed examples on how Theor. 14.3 must be instantiated for a meaningful application are given in Section 15 (lemma) and Section 16 (induction hypothesis), &c..

15 Example: Lemma Application

In this example we will keep to the following: The proofs are presented in the tableau calculus. As there are no inductive proofs, we omit the weight constructs completely. As no liberalized δ-rules are applied, the choice-conditions are always empty. As all proof trees have branching degree 1, we do not depict them. Assume that in the signature Σ we have the operator $*$, the constant $\mathsf{1}$, and the inverse function inv. We start with the empty proof forest $(\emptyset, \emptyset, \emptyset, \emptyset, \emptyset)$. We start the new proof tree number 1 for

$$x_1^{\forall,\mathrm{w}} * (y_1^{\forall,\mathrm{w}} * z_1^{\forall,\mathrm{w}}) = (x_1^{\forall,\mathrm{w}} * y_1^{\forall,\mathrm{w}}) * z_1^{\forall,\mathrm{w}} \tag{1}$$

by a Hypothesizing step in the tableau calculus of Definition 13.4, just as two new proof trees for

$$\mathsf{1} * x_2^{\forall,\mathrm{w}} = x_2^{\forall,\mathrm{w}} \tag{2}$$
$$\mathsf{inv}(x_3^{\forall,\mathrm{w}}) * x_3^{\forall,\mathrm{w}} = \mathsf{1}. \tag{3}$$

With these three trees we have the axioms of group theory at hand via lemma application. Now we really want to prove something. We start the new proof tree number 4 for

$$\forall x.\ \ x * \mathsf{inv}(x) = \mathsf{1} \tag{4}$$

by a Hypothesizing step. The the root of proof tree 4 is labeled with

$$\neg\forall x.\ \ x * \mathsf{inv}(x) = \mathsf{1}$$

A δ-step (cf. Theor. 14.1) adds the child

$$x_4^{\forall,\mathrm{w}} * \mathsf{inv}(x_4^{\forall,\mathrm{w}}) \neq \mathsf{1}$$

Our variable-condition is still empty because no free variables occur in $x * \mathsf{inv}(x) = \mathsf{1}$.
Applying the sequent of proof tree 3 in the way of Theor. 14.3 adds the new child

$$\mathsf{inv}(x_1^{\exists}) * x_1^{\exists} = \mathsf{1}$$

to proof tree 4 and inserts the pair $(3, 4)$ into our lemma application relation. A Rewrite step (cf. Theor. 14.1) with this equality from right to left produces the new child

$$x_4^{\forall,\mathrm{w}} * \mathsf{inv}(x_4^{\forall,\mathrm{w}}) \neq \mathsf{inv}(x_1^{\exists}) * x_1^{\exists}$$

Applying the sequent of proof tree 2 in the way of Theor. 14.3 adds the new child

$$\mathsf{1} * x_2^{\exists} = x_2^{\exists}$$

to proof tree 4 and inserts the pair $(2, 4)$ into our lemma application relation.
An Instantiation step (cf. Definition 13.4) transforms this child into

$$\mathsf{1} * x_1^{\exists} = x_1^{\exists}$$

which can be used for a Rewrite step from right to left adding the child

$$x_4^{\forall,\mathrm{w}} * \mathsf{inv}(x_4^{\forall,\mathrm{w}}) \neq \mathsf{inv}(x_1^{\exists}) * (\mathsf{1} * x_1^{\exists})$$

Now our variable-condition is $R := \{(x_1^{\exists}, x_2^{\exists})\}$. Since $x_2^{\exists}$ does not occur anywhere else in our current proof forest, this R does not put any restrictions on existential R-substitutions, unless we re-use $x_2^{\exists}$.[21] Therefore, in future applications of Theor. 14.3 we will not introduce new free existential variables at all when we already know how to instantiate them.
Applying the sequent of proof tree 3 in the way of Theor. 14.3 adds the new child

$$\mathsf{inv}(x_3^{\exists}) * x_3^{\exists} = \mathsf{1}$$

A Rewrite step (cf. Theor. 14.1) with this equality from right to left produces the new child

$$x_4^{\forall,\mathrm{w}} * \mathsf{inv}(x_4^{\forall,\mathrm{w}}) \neq \mathsf{inv}(x_1^{\exists}) * ((\mathsf{inv}(x_3^{\exists}) * x_3^{\exists}) * x_1^{\exists})$$

With two applications of the sequent of proof tree 1, this can be rewritten into

$$x_4^{\forall,\mathrm{w}} * \mathsf{inv}(x_4^{\forall,\mathrm{w}}) \neq (\mathsf{inv}(x_1^{\exists}) * \mathsf{inv}(x_3^{\exists})) * (x_3^{\exists} * x_1^{\exists})$$

Note that our lemma application relation now is $\{1, 2, 3\} \times \{4\}$.
Applying the sequent of proof tree 3 in the way of Theor. 14.3 adds the new child

$$\mathsf{inv}(x_4^{\exists}) * x_4^{\exists} = 1$$

In order to use this for a Rewrite step from left to right we apply the unifier $\sigma :=$ $\{x_1^{\exists} \mapsto \mathsf{inv}(x_3^{\exists}),\ x_4^{\exists} \mapsto \mathsf{inv}(x_3^{\exists})\}$ to the whole proof forest and—after the Rewrite step—get the new child

$$x_4^{\forall,\mathrm{w}} * \mathsf{inv}(x_4^{\forall,\mathrm{w}}) \neq 1 * (x_3^{\exists} * \mathsf{inv}(x_3^{\exists}))$$

Note that this unifier is an existential R-substitution and that the σ-update of R is given by

$$\begin{tikzcd} x_3^{\exists} \arrow[r, "E_\sigma"'] \arrow[dr, "E_\sigma"'] & x_1^{\exists} \arrow[r, "R"'] & x_2^{\exists} \\ & x_4^{\exists} & \end{tikzcd}$$

which still puts no restrictions whatsoever on existential R-substitutions of the free universal variables that are still present in our tree.[22]

With an application of the sequent of proof tree 2, this can be rewritten into

$$x_4^{\forall,\mathrm{w}} * \mathsf{inv}(x_4^{\forall,\mathrm{w}}) \neq x_3^{\exists} * \mathsf{inv}(x_3^{\exists})$$

An Instantiation step produces the new child

$$x_4^{\forall,\mathrm{w}} * \mathsf{inv}(x_4^{\forall,\mathrm{w}}) \neq x_4^{\forall,\mathrm{w}} * \mathsf{inv}(x_4^{\forall,\mathrm{w}})$$

Now the tree is closed because all sequents of the form $(t = t)\,\Lambda$ are assumed to be in our axioms $\mathcal{AX}$. By Theor. 13.6 we now know that $\forall x.\ \ x * \mathsf{inv}(x) = 1$ is $\emptyset$-valid, provided that the proof trees 1, 2, and 3 are closed, which is the case when we assume their sequents to be in $\mathcal{AX}$.

Now we start proof tree 5 for

$$x_5^{\forall,\mathrm{w}} * \mathsf{inv}(x_5^{\forall,\mathrm{w}}) = 1 \tag{5}$$

by a Hypothesizing step. Note that the sequent is not really different from that of proof tree 4. In order to have only one version of each lemma and to know how it looks like, we should make up our minds for one of the two forms. We do prefer the form of proof tree 5 because it will be more useful for ITP. For deductive theorem proving, the two only differ in that the form of proof tree 5 is handier for lemma application. To see this, we will prove each with the help of the other.

A lemma application according to Theor. 14.3 of the sequent of proof tree 4 to proof tree 5 whose root is labeled with $x_5^{\forall,\mathrm{w}} * \mathsf{inv}(x_5^{\forall,\mathrm{w}}) \neq 1$ adds the child $\forall x.\ \ x * \mathsf{inv}(x) = 1$.

A γ-step adds the child $x_5^{\exists} * \mathsf{inv}(x_5^{\exists}) = 1$.

An Instantiation step transforms this into $x_5^{\forall,\mathrm{w}} * \mathsf{inv}(x_5^{\forall,\mathrm{w}}) = 1$. Now proof tree 5 is closed because all sequents of the form $A\,\Lambda\,\overline{A}$ are assumed to be in our axioms $\mathcal{AX}$.

Finally, we start another proof tree number 6 for the sequent of proof tree 4. The root is again labeled with

$$\neg\forall x.\ \ x * \mathsf{inv}(x) = 1$$

A δ-step adds the child

$$x_6^{\forall,\mathrm{w}} * \mathsf{inv}(x_6^{\forall,\mathrm{w}}) \neq 1$$

Applying the sequent of proof tree 5 in the way of Theor. 14.3 adds the new child

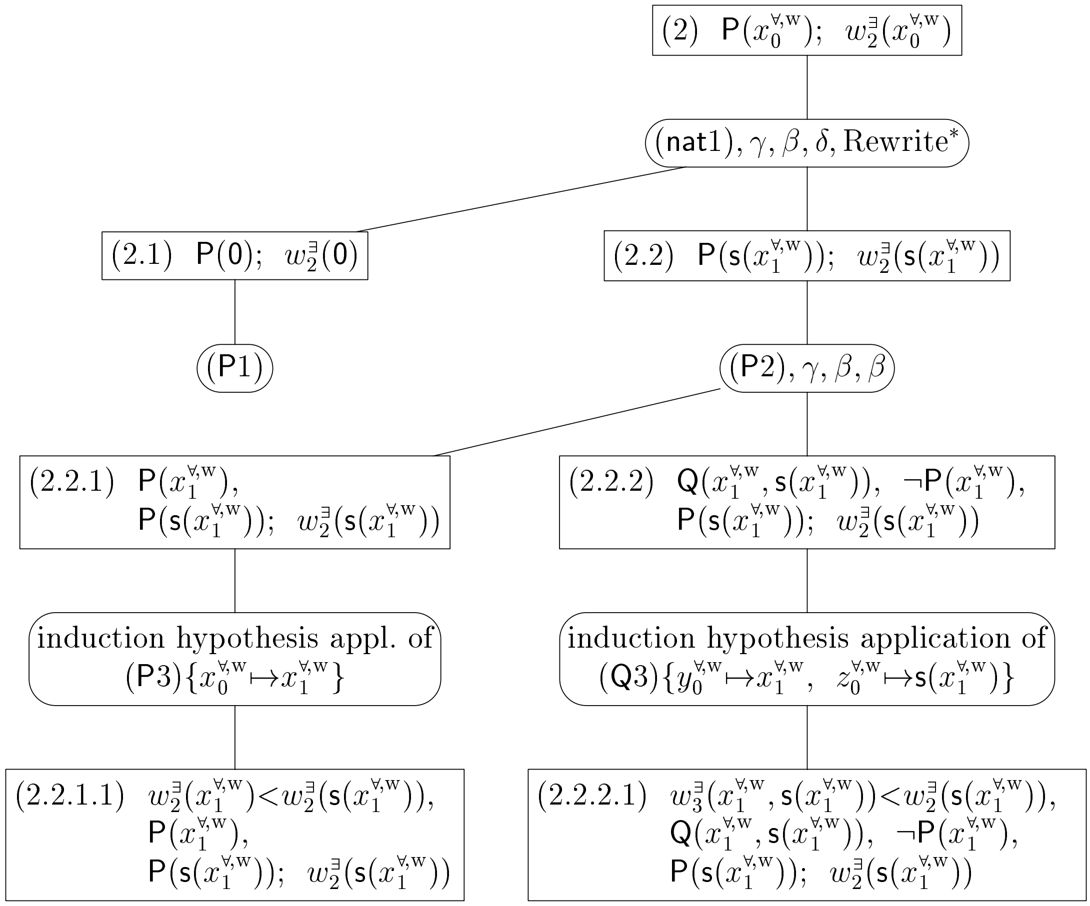
$$x_6^{\exists} * \mathsf{inv}(x_6^{\exists}) = 1$$

An Instantiation step transforms this into

$$x_6^{\forall,\mathrm{w}} * \mathsf{inv}(x_6^{\forall,\mathrm{w}}) = 1$$

Now proof tree 6 is also closed because all sequents of the form $A\,\overline{A}\,\Lambda$ are assumed to be in our axioms $\mathcal{AX}$.

16 Example: Mutual Induction



The toy example of this section illustrates how mutual induction works in our framework. Note, however, that (due to mutual induction and non-trivial weights) even this toy example has no straightforward proofs in the ITP calculus of Baaz &al. (1997) or any known ITP system with the exception of QuodLibet. The signature consists of the signature presented in Section 4.1 and additionally contains the predicates $\mathsf{P} : \mathsf{nat} \to \mathsf{bool}$ and $\mathsf{Q} : \mathsf{nat} \to \mathsf{nat} \to \mathsf{bool}$. Besides the axiom $(\mathsf{nat1})$ of Section 4.1, we have the following axioms, defining the special predicates of our example.

$(\mathsf{P1})$		$\mathsf{P}(0)$
$(\mathsf{P2})$	$\forall x.$	$\big(\ \mathsf{P}(\mathsf{s}(x)) \Leftarrow \big(\ \mathsf{P}(x) \wedge \mathsf{Q}(x, \mathsf{s}(x))\ \big)\ \big)$
$(\mathsf{Q1})$	$\forall x.$	$\mathsf{Q}(x, 0)$
$(\mathsf{Q2})$	$\forall x, y.$	$\big(\ \mathsf{Q}(x, \mathsf{s}(y)) \Leftarrow \big(\ \mathsf{Q}(x, y) \wedge \mathsf{P}(x)\ \big)\ \big)$

We want to show that both predicates are tautological:

$(\mathsf{P3})$	$\mathsf{P}(x_0^{\forall,\mathrm{w}})$; $w_2^{\exists}(x_0^{\forall,\mathrm{w}})$
$(\mathsf{Q3})$	$\mathsf{Q}(y_0^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}})$; $w_3^{\exists}(y_0^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}})$

Recall that weight constructs in QuodLibet consist of weight terms only (like $w_2^{\exists}(x_0^{\forall,\mathrm{w}})$ in $(\mathsf{P3})$) because the induction ordering is a globally fixed constant.

- (3) $\mathsf{Q}(y_0^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}});\;\; w_3^{\exists}(y_0^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}})$
  - ($\mathsf{nat1}$), γ, β, δ, Rewrite*
    - (3.1) $\mathsf{Q}(y_0^{\forall,\mathrm{w}}, \mathsf{0});\;\; w_3^{\exists}(y_0^{\forall,\mathrm{w}}, \mathsf{0})$
      - ($\mathsf{Q1}$)
    - (3.2) $\mathsf{Q}(y_0^{\forall,\mathrm{w}}, \mathsf{s}(z_1^{\forall,\mathrm{w}}));\;\; w_3^{\exists}(y_0^{\forall,\mathrm{w}}, \mathsf{s}(z_1^{\forall,\mathrm{w}}))$
      - ($\mathsf{Q2}$), γ, β, β
        - (3.2.1) $\mathsf{Q}(y_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}),\;\; \mathsf{Q}(y_0^{\forall,\mathrm{w}}, \mathsf{s}(z_1^{\forall,\mathrm{w}}));\;\; w_3^{\exists}(y_0^{\forall,\mathrm{w}}, \mathsf{s}(z_1^{\forall,\mathrm{w}}))$
          - ind. hyp. appl. of ($\mathsf{Q3}$)$\{z_0^{\forall,\mathrm{w}} \mapsto z_1^{\forall,\mathrm{w}}\}$
            - (3.2.1.1) $w_3^{\exists}(y_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}) < w_3^{\exists}(y_0^{\forall,\mathrm{w}}, \mathsf{s}(z_1^{\forall,\mathrm{w}})),\;\; \mathsf{Q}(y_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}),\;\; \mathsf{Q}(y_0^{\forall,\mathrm{w}}, \mathsf{s}(z_1^{\forall,\mathrm{w}}));\;\; w_3^{\exists}(y_0^{\forall,\mathrm{w}}, \mathsf{s}(z_1^{\forall,\mathrm{w}}))$
        - (3.2.2) $\mathsf{P}(y_0^{\forall,\mathrm{w}}),\;\; \neg\mathsf{Q}(y_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}),\;\; \mathsf{Q}(y_0^{\forall,\mathrm{w}}, \mathsf{s}(z_1^{\forall,\mathrm{w}}));\;\; w_3^{\exists}(y_0^{\forall,\mathrm{w}}, \mathsf{s}(z_1^{\forall,\mathrm{w}}))$
          - ind. hyp. appl. of ($\mathsf{P3}$)$\{x_0^{\forall,\mathrm{w}} \mapsto y_0^{\forall,\mathrm{w}}\}$
            - (3.2.2.1) $w_2^{\exists}(y_0^{\forall,\mathrm{w}}) < w_3^{\exists}(y_0^{\forall,\mathrm{w}}, \mathsf{s}(z_1^{\forall,\mathrm{w}})),\;\; \mathsf{P}(y_0^{\forall,\mathrm{w}}),\;\; \neg\mathsf{Q}(y_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}),\;\; \mathsf{Q}(y_0^{\forall,\mathrm{w}}, \mathsf{s}(z_1^{\forall,\mathrm{w}}));\;\; w_3^{\exists}(y_0^{\forall,\mathrm{w}}, \mathsf{s}(z_1^{\forall,\mathrm{w}}))$

We start with the variable-condition

$$R \;:=\; \mathcal{V}_{\exists}(\mathsf{P3}) \times \mathcal{V}_{\forall,\mathrm{w}}(\mathsf{P3}) \cup \mathcal{V}_{\exists}(\mathsf{Q3}) \times \mathcal{V}_{\forall,\mathrm{w}}(\mathsf{Q3}) \;=\; \{w_2^{\exists}\} \times \{x_0^{\forall,\mathrm{w}}\} \cup \{w_3^{\exists}\} \times \{y_0^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}\}$$

in order to have all weak free universal variables of ($\mathsf{P3}$) or ($\mathsf{Q3}$) in the set Y of Theor. 14.3. After several inference steps XQuodLibet presents two partial sequent proof trees for ($\mathsf{P3}$) and ($\mathsf{Q3}$) similar to the first two trees depicted in this section.

Just as in Section 4.1, we have used the round-edged nodes to give some information how the inference steps can be achieved in terms of general inference rules as the ones presented in Theor. 14.1 and Theor. 14.3.

As the inferences applied to (2) do not differ from those applied to (1) in Section 4.1, let us have a closer look at the inference below (2.2).

The defining formula ($\mathsf{P2}$) is applied just like a lemma in Theor. 14.3, i.e. its single formula is added in negated form. Thus, the round-edged node labeled with "($\mathsf{P2}$), γ, β, β" can be replaced with the following subtree. Note that the γ-step actually first introduces a free existential variable that is then instantiated with $x_1^{\forall,\mathrm{w}}$ in order to close the leftmost leaf of the tree below, which then does not have to be presented anymore.

$$(2.2)$$

$$\neg\forall x.\ \big(\ \mathsf{P}(\mathsf{s}(x)) \Leftarrow \big(\ \mathsf{P}(x) \wedge \mathsf{Q}(x, \mathsf{s}(x))\ \big)\ \big),\\ \mathsf{P}(\mathsf{s}(x_1^{\forall,\mathrm{w}}));\ \ w_2^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}))$$

$$\neg\big(\ \mathsf{P}(\mathsf{s}(x_1^{\forall,\mathrm{w}})) \Leftarrow \big(\ \mathsf{P}(x_1^{\forall,\mathrm{w}}) \wedge \mathsf{Q}(x_1^{\forall,\mathrm{w}}, \mathsf{s}(x_1^{\forall,\mathrm{w}}))\ \big)\ \big),\\ \mathsf{P}(\mathsf{s}(x_1^{\forall,\mathrm{w}}));\ \ w_2^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}))$$

$$\neg\mathsf{P}(\mathsf{s}(x_1^{\forall,\mathrm{w}})),\\ \mathsf{P}(\mathsf{s}(x_1^{\forall,\mathrm{w}}));\ \ w_2^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}))$$

$$\big(\ \mathsf{P}(x_1^{\forall,\mathrm{w}}) \wedge \mathsf{Q}(x_1^{\forall,\mathrm{w}}, \mathsf{s}(x_1^{\forall,\mathrm{w}}))\ \big),\\ \mathsf{P}(\mathsf{s}(x_1^{\forall,\mathrm{w}}));\ \ w_2^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}))$$

$$(2.2.1) \qquad (2.2.2)$$

Even more interesting is what happens below (2.2.2). We instantiate the (meta-) variables of Theor. 14.3, in the following way: $\varPhi := \mathsf{Q}(y_0^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}})$, $\daleth := w_3^{\exists}(y_0^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}})$, $Y := \{y_0^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}\}$, $\varrho := \{y_0^{\forall,\mathrm{w}} \mapsto u^{\exists},\ z_0^{\forall,\mathrm{w}} \mapsto v^{\exists}\}$, $M := \{\neg\mathsf{Q}(u^{\exists}, v^{\exists}),\ w_3^{\exists}(u^{\exists}, v^{\exists}) < w_2^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}))\}$. This results in the tree below. When we instantiate with $\{u^{\exists} \mapsto x_1^{\forall,\mathrm{w}},\ v^{\exists} \mapsto \mathsf{s}(x_1^{\forall,\mathrm{w}})\}$ its left leaf gets closed and its right leaf becomes (2.2.2.1).

$$(2.2.2)$$

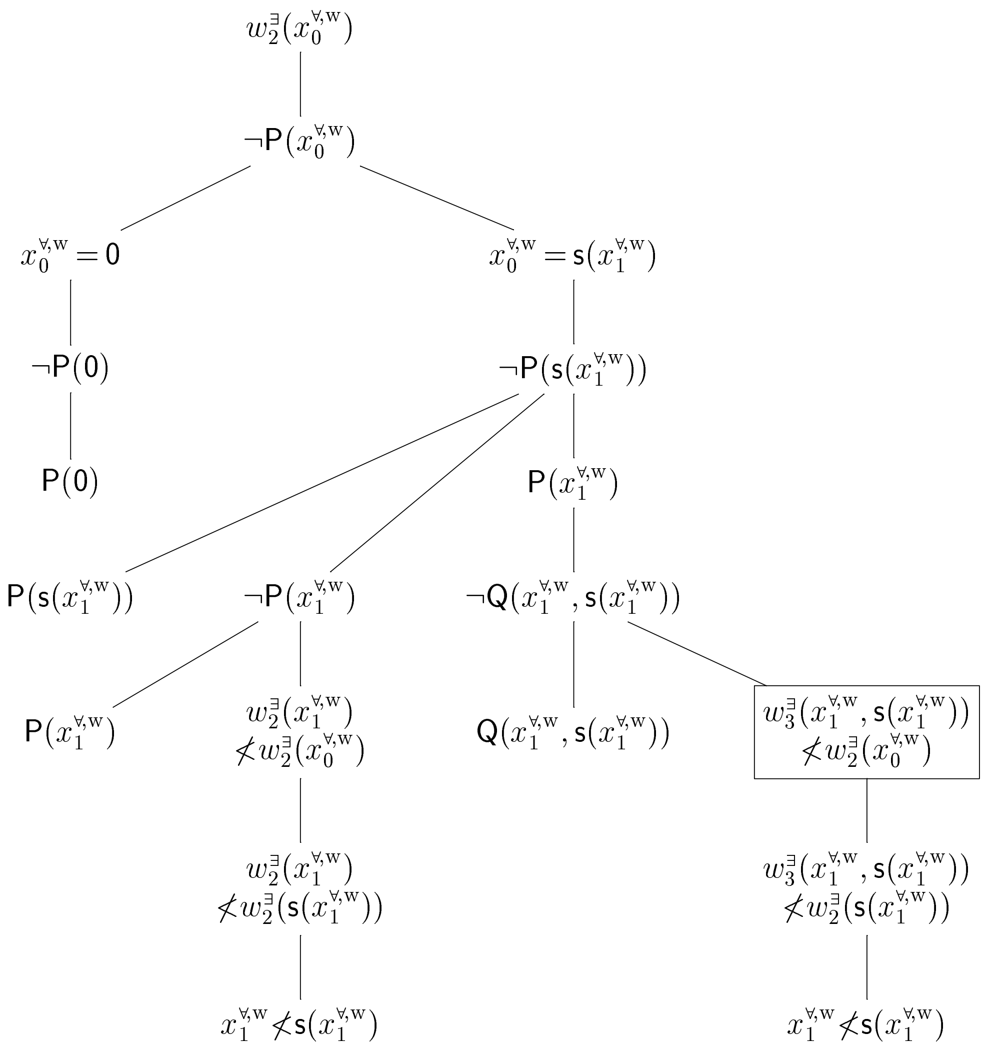
$$\neg\mathsf{Q}(u^{\exists}, v^{\exists}),\\ \mathsf{Q}(x_1^{\forall,\mathrm{w}}, \mathsf{s}(x_1^{\forall,\mathrm{w}})),\ \ \neg\mathsf{P}(x_1^{\forall,\mathrm{w}}),\\ \mathsf{P}(\mathsf{s}(x_1^{\forall,\mathrm{w}}));\ \ w_2^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}))$$

$$w_3^{\exists}(u^{\exists}, v^{\exists}) < w_2^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}})),\\ \mathsf{Q}(x_1^{\forall,\mathrm{w}}, \mathsf{s}(x_1^{\forall,\mathrm{w}})),\ \ \neg\mathsf{P}(x_1^{\forall,\mathrm{w}}),\\ \mathsf{P}(\mathsf{s}(x_1^{\forall,\mathrm{w}}));\ \ w_2^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}))$$

We have applied each of the two syntactical constructs ($\mathsf{P3}$) and ($\mathsf{Q3}$) in each of their two proof trees (2) and (3), resp.. Luckily we used induction hypothesis application instead of lemma application. The latter would have resulted in a lemma application relation of $\{2,3\} \times \{2,3\}$ which is not wellfounded and our proof trees would have been useless because we would never be able to apply Theor. 13.6. As we have used induction hypothesis application instead of lemma application, we have produced the four additional leaves (2.2.1.1), (2.2.2.1), (3.2.1.1), and (3.2.2.1), which are still open. We choose our 2nd order weight functions according to $w_2^{\exists}(x) := (x)$ and $w_3^{\exists}(x,y) := (x,y)$, using the lexicographic combination of Section 4.2.[23] Now the proof attempt can be successfully completed: E.g., the first literal of (2.2.1.1) turns into $(x_1^{\forall,\mathrm{w}}, \mathsf{0}) < (\mathsf{s}(x_1^{\forall,\mathrm{w}}), \mathsf{0})$, which after applying (<4) of Note 13, in a γ- and a β-step reads $x_1^{\forall,\mathrm{w}} < \mathsf{s}(x_1^{\forall,\mathrm{w}})$ which is an ordering axiom in QuodLibet, as explained in Note 13.

Finally in this section we should answer the following question: *Which steps in this proof were typical for ITP in the sense that their soundness relies on notions of inductive validity* *instead of the stronger notion of deductive validity?* Besides the four induction hypothesis applications, the final branch closure rules for $<$-literals are typical for induction because they require that, in all models in K, the successor of each natural number is different from that natural number and each natural number is built-up from zero by a finite number of successor steps (i.e. there are neither cycles nor **Z**-chains in the models, cf. Enderton (1973)).

17 Sequents versus Tableaus in ITP



In this section we are going to compare the appropriateness of sequent versus tableau calculi under the special aspect of ITP. To this end we first see what the sequent calculus proof of Section 16 would look like in a tableau calculus. After the first Hypothesizing step, the initial tableau for ($\mathsf{P3}$) looks the following way.

$$\begin{array}{c} w_2^{\exists}(x_0^{\forall,\mathrm{w}}) \\ | \\ \neg\mathsf{P}(x_0^{\forall,\mathrm{w}}) \end{array}$$

Note that this differs from (P3) in duality. While this is not a hindrance for completely automatic ITP systems, it poses considerable practical problems in systems where user-guidance is possible: The primitive process of switching duality is a typical source of errors for human beings (or me at least).

For the closed complete proof tree for (P3) we have chosen a representation according to clausal tableau calculi because there is not enough space for non-atomic formulas here.

Let us have a closer look at the boxed formula in the tableau. It results from induction hypothesis application of (Q3). Note that the only difference to an Extension step in Model Elimination tableaus (cf. Baumgartner &al. (1997)) lies with the additional child (the boxed node), which asks us to show that the instance of the hypothesis is smaller than the weight of our proof tree. Indeed: As the induction ordering is fixed here, hypothesis application differs from the standard lemma (or axiom) application only in producing an additional $<$-goal. This makes hypothesis application a little more expensive than lemma application. The left-hand term $w_3^{\exists}(x_1^{\forall,\mathrm{w}}, \mathsf{s}(x_1^{\forall,\mathrm{w}}))$ is the weight term of (Q3) instantiated via $\{y_0^{\forall,\mathrm{w}} {\mapsto} \mathsf{s}(x_1^{\forall,\mathrm{w}}),\ z_0^{\forall,\mathrm{w}} {\mapsto} x_1^{\forall,\mathrm{w}}\}$ because this substitution enables the left sibling of the boxed node to close its branch with the instantiated negated formula of (Q3). The right-hand term $w_2^{\exists}(x_0^{\forall,\mathrm{w}})$ comes down from the root of the tree. Contrary to the sequent calculus example where the weight of the root is carried along and updated on its way down, we have to rewrite the variable $x_0^{\forall,\mathrm{w}}$ in it with an ancestor equality literal in order to know what the root weight means in the local context.

Of course, there is an analogous closed tableau for (Q3) because otherwise Theor. 13.6 would not allow us to conclude that $\mathsf{P}(x_0^{\forall,\mathrm{w}})$ is valid.

Note that the sequent calculus proof tree is not equal to the result of the standard transformation of the tableau tree. The standard transformation of a tableau tree into a sequent tree works for inductive trees just as for deductive trees:

1. Bottom-up replace the label of each node with the syntactical construct listing the conjugates of the formulas and the weight labeling the (partial) branch from this node to the root.
2. Remove the root part of the tree where the nodes are ancestors of a node of the initial Hypothesizing step (in our example: remove the root node).

This standard transformation multiplies the number of formulas labeling each proof tree with at most nearly the depth of that tree, but does not use the advantages of sequent calculi, namely the ability to simplify formulas that label ancestor nodes in a tableau calculus. E.g., in the tableau tree it is not possible to rewrite the literal $\neg\mathsf{P}(x_0^{\forall,\mathrm{w}})$ with the equality literals below it in place. In tableau trees, an equality literal can be used to rewrite formulas of its offspring in place, whereas it must copy ancestor formulas beforehand down to its offspring because the ancestor is also part of other branches that do not include the equality literal. Moreover, the weight term can be rewritten in the sequent tree, which again is not possible in the tableau version where the weight is at the root node. Since $x_0^{\forall,\mathrm{w}}$ is in solved form after the Rewrite steps, we know that validity cannot rely on the equality literals containing it. This means that we can safely remove both equality literals in the sequent tree so that they do not appear in (2.1) and (2.2). Removing redundant formulas is the most important simplification step besides contextual rewriting. This is impossible in tableau trees unless the redundancy of the formula is due to the ancestor nodes only, which is the case only for useless formulas that should not have been added at all.

Note that formulas like ($\mathsf{nat1}$) from Section 4.1 make equality omnipresent in ITP and that these simplification steps are even more important in inductive than in deductive theorem proving: Not only do they play a role in the generation of appropriate induction hypotheses; in addition to the detection of invalid input theorems they are an essential part of the failure detection process that has to compensate for *over-generalization* of induction hypotheses: Indeed, ITP often is only successful when one tries to show theorems that are more general than the ones one initially intended to show. This is because an inductive theorem is not only a task (as goal) but also a tool (as induction hypothesis) for ITP. This generalization is *unsafe* in the sense that it may over-generalize a valid hypothesis into an invalid one. Therefore, generalization should not be modeled in Expansion steps within a tree. Instead, the generalized sequent should start a new tree (Hypothesizing step) and be later applied to the original tree as a lemma or an induction hypothesis. Since even a valid input theorem may result in an invalid goal due to over-generalization, the ability of an ITP system to detect invalid goals is of major importance in practice, cf. Section 13.2.

In Wirth (1997) and in QuodLibet the Expansion from (2) into (2.1) and (2.2) is done in a single inference step called "substitution add" applying a "covering set of substitutions". Note that the state of the sequent proof resulting from this step is much simpler than the corresponding state of the tableau proof. The former consists of the nodes (2.1) and (2.2) and has two formulas and one variable. The latter consists of a six node tree with five formulas and two variables. This is of practical importance because tactics for proof search are more easily confused with less concise proof state representations. The rest of the whole sequent proof is analogous to the tableau proof with the exception that all rewrite steps of the tableau tree are omitted since there are no equality literals to rewrite with and the terms are already in normal form.

Another possibility restricted to sequent calculi is that each syntactical construct labeling a node in the trees can be applied as an induction hypothesis. We do not see a real advantage in this because splitting the tree in two above such an induction hypothesis results in a better structure of the proof forest and in more successful proofs because we can adjust the syntactical construct appropriately: Suppose we had not started a new proof tree for the hypothesis for Q but instead kept the hypothesis for Q down in the tree (2) at position (2.2.2). Several unsafe generalization steps would have been necessary before

$$\mathsf{Q}(x_1^{\forall,\mathrm{w}}, \mathsf{s}(x_1^{\forall,\mathrm{w}})),\ \ \neg\mathsf{P}(x_1^{\forall,\mathrm{w}}),\ \ \mathsf{P}(\mathsf{s}(x_1^{\forall,\mathrm{w}}));\ \ w_0^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}))$$

would have become useful as an induction hypothesis, namely removing the second and third formula, generalizing $\mathsf{s}(x_1^{\forall,\mathrm{w}})$ to a new variable, and switching to a weight that measures also this new variable.

Moreover, in practice one should not apply the hypothesis for Q in the tree for P before it is obvious that the tree for Q mutually needs the hypothesis for P: Most of the time a proof for Q can be completed in a proof forest not containing the tree for P. In this case, not only the number of trees in the proof forest for Q gets smaller, but also the tree for P because ($\mathsf{Q}3$) can then be applied as a lemma and not as an induction hypothesis, which cuts off the rightmost $<$-branch of the proof tree of P.

18 Example: Eager Hypotheses Generation

Let us try to find a lower bound for the Ackermann function $\mathsf{ack} : \mathsf{nat} \to \mathsf{nat} \to \mathsf{nat}$ w.r.t. the ordering on natural numbers $\mathsf{less} : \mathsf{nat} \to \mathsf{nat} \to \mathsf{bool}$, assuming the following axioms.

(ack1)	$\forall y.$	$\mathsf{ack}(0, y){=}\mathsf{s}(y)$
(ack2)	$\forall x, y.$	$\mathsf{ack}(\mathsf{s}(x), 0){=}\mathsf{ack}(x, \mathsf{s}(0))$
(ack3)	$\forall x, y.$	$\mathsf{ack}(\mathsf{s}(x), \mathsf{s}(y)){=}\mathsf{ack}(x, \mathsf{ack}(\mathsf{s}(x), y))$
(less1)	$\forall y.$	$\mathsf{less}(0, \mathsf{s}(y)){=}\mathsf{true}$
(less2)	$\forall x.$	$\mathsf{less}(x, 0){=}\mathsf{false}$
(less3)	$\forall x, y.$	$\mathsf{less}(\mathsf{s}(x), \mathsf{s}(y)){=}\mathsf{less}(x, y)$

The standard lemmas for less that have very simple inductive proofs in QuodLibet are:

(less4)	$\forall x.$	$\mathsf{less}(x, \mathsf{s}(x))$
(less5)	$\forall x, y.$	$\left(\ \mathsf{less}(x, y) \Rightarrow \mathsf{less}(x, \mathsf{s}(y))\ \right)$
(less6)	$\forall x, y.$	$\left(\ \mathsf{less}(\mathsf{s}(x), y) \Rightarrow \mathsf{less}(x, y)\ \right)$
(less7)	$\forall x, y, z.$	$\left(\left(\begin{array}{rl} & \mathsf{less}(x, y) \\ \wedge & \mathsf{less}(y, z) \end{array} \right) \Rightarrow \mathsf{less}(\mathsf{s}(x), z) \right)$

Note that for Boolean terms t we abbreviate the equation $t{=}\mathsf{true}$ with t. Moreover, note that (less7) is a strengthened version of transitivity. The simple transitivity is a simple consequence of it, using (less6).

Let us start with a Hypothesizing step in the sequent calculus of Definition 13.4, posing the query for a lower bound $z_0^{\exists} : \mathsf{nat} \to \mathsf{nat} \to \mathsf{nat}$

(4) $\mathsf{less}(z_0^{\exists}(x_0^{\forall,\mathrm{w}}, y_0^{\forall,\mathrm{w}}), \mathsf{ack}(x_0^{\forall,\mathrm{w}}, y_0^{\forall,\mathrm{w}}));\ \ w_4^{\exists}(x_0^{\forall,\mathrm{w}}, y_0^{\forall,\mathrm{w}})$

with variable-condition $R := \{z_0^{\exists}, w_4^{\exists}\} \times \{x_0^{\forall,\mathrm{w}}, y_0^{\forall,\mathrm{w}}\}$.

Note that we have to let $z_0^{\exists}$ be a higher-order variable: If $z_0^{\exists}$ were a first-order variable, it could not depend on $x_0^{\forall,\mathrm{w}}$ and $y_0^{\forall,\mathrm{w}}$ due to R, resulting in a constant lower bound that would not be too interesting. If we did not include $z_0^{\exists}$ into $\mathrm{dom}(R)$, however, Theor. 14.3 would not permit us to do induction on $x_0^{\forall,\mathrm{w}}$ and $y_0^{\forall,\mathrm{w}}$ because they would not be elements of the set Y.

Applying (nat1) (cf. Section 4.1) as a lemma according to Theor. 14.3 yields the two goals

(4.1) $\mathsf{less}(z_0^{\exists}(0, y_0^{\forall,\mathrm{w}}), \mathsf{ack}(0, y_0^{\forall,\mathrm{w}}));\ \ w_4^{\exists}(0, y_0^{\forall,\mathrm{w}})$
(4.2) $\mathsf{less}(z_0^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}), y_0^{\forall,\mathrm{w}}), \mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}), y_0^{\forall,\mathrm{w}}));\ \ w_4^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}), y_0^{\forall,\mathrm{w}})$

just it was as it was explained at the end of Section 4.1, adding $\{z_0^{\exists}, w_4^{\exists}\} \times \{x_1^{\forall,\mathrm{w}}\}$ to the variable-condition. The same procedure again yields

(4.2.1) $\mathsf{less}(z_0^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}), 0), \mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}), 0));\ \ w_4^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}), 0)$
(4.2.2) $\mathsf{less}(z_0^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}), \mathsf{s}(y_1^{\forall,\mathrm{w}})), \mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}), \mathsf{s}(y_1^{\forall,\mathrm{w}})));\ \ w_4^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}), \mathsf{s}(y_1^{\forall,\mathrm{w}}))$

adding $\{z_0^{\exists}, w_4^{\exists}\} \times \{y_1^{\forall,\mathrm{w}}\}$ to the variable-condition.

Rewriting (4.1), (4.2.1), and (4.2.2) with (ack1), (ack2), and (ack3), resp., yields

(4.1.1) $\mathsf{less}(z_0^{\exists}(0, y_0^{\forall,\mathrm{w}}), \mathsf{s}(y_0^{\forall,\mathrm{w}}));\ \ w_4^{\exists}(0, y_0^{\forall,\mathrm{w}})$
(4.2.1.1) $\mathsf{less}(z_0^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}), 0), \mathsf{ack}(x_1^{\forall,\mathrm{w}}, \mathsf{s}(0)));\ \ w_4^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}), 0)$
(4.2.2.1) $\mathsf{less}(z_0^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}), \mathsf{s}(y_1^{\forall,\mathrm{w}})), \mathsf{ack}(x_1^{\forall,\mathrm{w}}, \mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}), y_1^{\forall,\mathrm{w}})));\ \ w_4^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}), \mathsf{s}(y_1^{\forall,\mathrm{w}}))$

In our previous examples the generation of induction hypotheses was always lazy in the sense of Protzen (1994). In this case, however, in order to be able to use goal-directedness also w.r.t. the induction hypotheses, we should generate them eagerly in the way suggested by the recursion analysis of explicit induction, cf. e.g. Boyer & Moore (1979), Walther (1992).

Recursion analysis and eager hypotheses generation are very useful for finding simple proofs completely automatically. Although the ITP system NQTHM (cf. Boyer & Moore (1988)) cannot accept (4) because it does not have any free existential variables (not even existential quantification), if we instantiate (4) with the proper lower bound, NQTHM proves (4) completely automatically, even when the lemma (less7) is not provided and the function 'less' is redefined so that the built-in features for treating arithmetic cannot help. Moreover, during this proof the fascinating NQTHM guesses (less7) completely automatically using the goal-directedness w.r.t. the eagerly generated induction hypotheses. Indeed, if the eagerly generated induction hypotheses happen to be the right ones, they can help us to find missing lemmas or to find proper instantiations for free existential variables. Since it is folklore heuristic knowledge in ITP that a strong lower bound is often found by first finding a weaker one and then improving it, we should not look for an optimal lower bound with difficult proof but for a reasonable lower bound with simple proof.

Note that eager hypotheses generation is not possible with the induction rules of Baaz &al. (1997).

In our example, the induction hypotheses suggested for (4.2.1.1) and (4.2.2.1) result from matching the ack-subterm of (4) to the ack-subterms of (4.2.1.1) and (4.2.2.1). For (4.2.1.1) we get the substitution $\{x_0^{\forall,\mathrm{w}}\mapsto x_1^{\forall,\mathrm{w}},\ \ y_0^{\forall,\mathrm{w}}\mapsto\mathsf{s}(0)\}$ and for (4.2.2.1) the substitutions $\{x_0^{\forall,\mathrm{w}}\mapsto x_1^{\forall,\mathrm{w}},\ \ y_0^{\forall,\mathrm{w}}\mapsto\mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),y_1^{\forall,\mathrm{w}})\}$ and $\{x_0^{\forall,\mathrm{w}}\mapsto\mathsf{s}(x_1^{\forall,\mathrm{w}}),\ \ y_0^{\forall,\mathrm{w}}\mapsto y_1^{\forall,\mathrm{w}}\}$ resulting in:

(4.2.1.1.1) $\neg\mathsf{less}(z_0^{\exists}(x_1^{\forall,\mathrm{w}},\mathsf{s}(0)),\mathsf{ack}(x_1^{\forall,\mathrm{w}},\mathsf{s}(0))),\ \ \mathsf{less}(z_0^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),0),\mathsf{ack}(x_1^{\forall,\mathrm{w}},\mathsf{s}(0)));\ \ w_4^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),0)$
(4.2.1.1.2) $w_4^{\exists}(x_1^{\forall,\mathrm{w}},\mathsf{s}(0))<w_4^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),0),\ \ \mathsf{less}(z_0^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),0),\mathsf{ack}(x_1^{\forall,\mathrm{w}},\mathsf{s}(0)));\ \ w_4^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),0)$
(4.2.2.1.1) $\neg\mathsf{less}(z_0^{\exists}(x_1^{\forall,\mathrm{w}},\mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),y_1^{\forall,\mathrm{w}})),\mathsf{ack}(x_1^{\forall,\mathrm{w}},\mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),y_1^{\forall,\mathrm{w}}))),$
$\mathsf{less}(z_0^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),\mathsf{s}(y_1^{\forall,\mathrm{w}})),\mathsf{ack}(x_1^{\forall,\mathrm{w}},\mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),y_1^{\forall,\mathrm{w}})));\ \ w_4^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),\mathsf{s}(y_1^{\forall,\mathrm{w}}))$
(4.2.2.1.2) $w_4^{\exists}(x_1^{\forall,\mathrm{w}},\mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),y_1^{\forall,\mathrm{w}}))<w_4^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),\mathsf{s}(y_1^{\forall,\mathrm{w}})),\ \ \ldots$
(4.2.2.1.1.1) $\neg\mathsf{less}(z_0^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),y_1^{\forall,\mathrm{w}}),\mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),y_1^{\forall,\mathrm{w}})),$
$\neg\mathsf{less}(z_0^{\exists}(x_1^{\forall,\mathrm{w}},\mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),y_1^{\forall,\mathrm{w}})),\mathsf{ack}(x_1^{\forall,\mathrm{w}},\mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),y_1^{\forall,\mathrm{w}}))),$
$\mathsf{less}(z_0^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),\mathsf{s}(y_1^{\forall,\mathrm{w}})),\mathsf{ack}(x_1^{\forall,\mathrm{w}},\mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),y_1^{\forall,\mathrm{w}})));\ \ w_4^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),\mathsf{s}(y_1^{\forall,\mathrm{w}}))$
(4.2.2.1.1.2) $w_4^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),y_1^{\forall,\mathrm{w}})<w_4^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),\mathsf{s}(y_1^{\forall,\mathrm{w}})),\ \ \ldots$

After setting $w_4^{\exists}(x,y) := (x,y)$, the goals (4.2.1.1.2), (4.2.2.1.2), and (4.2.2.1.1.2) can be closed due to their first formulas. The whole proof up to now is the "eager induction hypotheses generation" suggested by recursion analysis of (4).

Now, (4.2.2.1.1.1) cries for a lemma application of (less7). Indeed, the lemma can close it, provided that we can identify the pairs $(\mathsf{s}(z_0^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),y_1^{\forall,\mathrm{w}})),z_0^{\exists}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),\mathsf{s}(y_1^{\forall,\mathrm{w}})))$ and $(\mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),y_1^{\forall,\mathrm{w}}),z_0^{\exists}(x_1^{\forall,\mathrm{w}},\mathsf{ack}(\mathsf{s}(x_1^{\forall,\mathrm{w}}),y_1^{\forall,\mathrm{w}})))$, which is achieved by their most general $\lambda\beta$-unifier, the projection $z_0^{\exists}(x,y) := y$.

Now (4.1.1) reads

(4.1.1') $\mathsf{less}(y_0^{\forall,\mathrm{w}},\mathsf{s}(y_0^{\forall,\mathrm{w}}));\ \ (0,y_0^{\forall,\mathrm{w}})$

which can be closed by an application of lemma (less4).

The only branch that is still open is

(4.2.1.1.1') $\neg\mathsf{less}(\mathsf{s}(0), \mathsf{ack}(x_1^{\forall,\mathrm{w}}, \mathsf{s}(0))),\ \ \mathsf{less}(0, \mathsf{ack}(x_1^{\forall,\mathrm{w}}, \mathsf{s}(0)));\ \ (\mathsf{s}(x_1^{\forall,\mathrm{w}}), 0)$

which can be closed by an application of lemma (less6).

This completes the proof of (4) with the answer that $z_0^{\exists}$ can be the projection to its second argument, i.e. the lower bound is $y_0^{\forall,\mathrm{w}}$.

Note that it is possible to do this proof with the first-order system QuodLibet because it is so lazy that one can use a symbol for an undefined function instead of the 2^{nd} order variable $z_0^{\exists}$. There is no 2^{nd} order unification but the user can set this function to be the projection during the proof. Since QuodLibet guarantees consistency of the specification (i.e. the existence of models where semantical equality of constructor ground terms implies syntactical equality) (provided arithmetic is consistent, cf. Gentzen (1938)) and admits partially defined and non-terminating functions, the actual proof in QuodLibet differs from the presented one by some additional subgoals that can be closed by a lemma stating that ack is a total function, which has a simple inductive proof. For the details cf. Kühler & Wirth (1996).

19 Example: Variable Induction Ordering

$\mathsf{0}$	:	nat
s	:	$\mathsf{nat} \to \mathsf{nat}$
$\mathsf{true},\ \mathsf{false}$	:	bool
$*,\ \mathsf{Rev}$	:	$(\mathsf{A} \to \mathsf{A} \to \mathsf{bool}) \to \mathsf{A} \to \mathsf{A} \to \mathsf{bool}$
Union	:	$(\mathsf{A} \to \mathsf{A} \to \mathsf{bool}) \to (\mathsf{A} \to \mathsf{A} \to \mathsf{bool}) \to \mathsf{A} \to \mathsf{A} \to \mathsf{bool}$
$\mathsf{Comm},\ \mathsf{LComm}$	:	$(\mathsf{A} \to \mathsf{A} \to \mathsf{bool}) \to (\mathsf{A} \to \mathsf{A} \to \mathsf{bool}) \to \mathsf{bool}$
Wellf	:	$(\mathsf{A} \to \mathsf{A} \to \mathsf{bool}) \to \mathsf{bool}$

In this section we are going to prove a generalized version of a lemma of M. H. A. Newman, namely that local commutation of two relations implies their commutation, provided that the reverse of their union is wellfounded.

Our displayed simply-typed higher-order signature is used to denote the following: $*(\longrightarrow)$ contains the transitive closure of the binary relation $\longrightarrow$ on A, $\mathsf{Rev}(\longrightarrow)$ is its reverse relation, and $\mathsf{Union}(\longrightarrow, \longrightarrow')$ is its union with $\longrightarrow'$. For all our Boolean terms t we abbreviate the equation $t{=}\mathsf{true}$ with t. For $\longrightarrow : \mathsf{A} \to \mathsf{A} \to \mathsf{bool}$, instead of $\longrightarrow(x, y)$ we write $x \longrightarrow y$, instead of $*(\longrightarrow, x, y)$ we write $x \overset{*}{\longrightarrow} y$, and instead of $\mathsf{Union}(\longrightarrow, \longrightarrow')$ we write $\longrightarrow \cup \longrightarrow'$.

$$(*1) \qquad \forall \longrightarrow, x, z.\ \left(x \overset{*}{\longrightarrow} z \Leftrightarrow \left(\begin{array}{c} x{=}z \\ \vee\ \ \exists y.\ \left(\begin{array}{c} x \longrightarrow y \\ \wedge\ \ y \overset{*}{\longrightarrow} z \end{array} \right) \end{array} \right) \right)$$

$$(\mathsf{Union1}) \qquad \forall \longrightarrow, \longrightarrow', x, y.\ \left(x(\longrightarrow \cup \longrightarrow')y \Leftrightarrow \left(\begin{array}{c} x \longrightarrow y \\ \vee\ \ x \longrightarrow' y \end{array} \right) \right)$$

$$(\mathsf{Rev1}) \qquad \forall \longrightarrow, x, y.\ \big(\mathsf{Rev}(\longrightarrow, x, y) \Leftrightarrow y \longrightarrow x \big)$$

($\mathsf{Comm1}$), ($\mathsf{LComm1}$), and ($\mathsf{Wellf1}$) are the properties of commutation, local commutation, and wellfoundedness, resp.:

$$(\mathsf{Comm1}) \qquad \forall \longrightarrow_0, \longrightarrow_1.\ \left(\begin{array}{l} \mathsf{Comm}(\longrightarrow_0, \longrightarrow_1) \\ \Leftrightarrow\ \ \forall x, y_0, y_1.\ \left(\begin{array}{l} \left(\begin{array}{c} x \overset{*}{\longrightarrow}_0 y_0 \\ \wedge\ \ x \overset{*}{\longrightarrow}_1 y_1 \end{array} \right) \\ \Rightarrow\ \ \exists z.\ \left(\begin{array}{c} y_0 \overset{*}{\longrightarrow}_1 z \\ \wedge\ \ y_1 \overset{*}{\longrightarrow}_0 z \end{array} \right) \end{array} \right) \end{array} \right)$$

$$(\mathsf{LComm1}) \qquad \forall \longrightarrow_0, \longrightarrow_1.\ \left(\begin{array}{l} \mathsf{LComm}(\longrightarrow_0, \longrightarrow_1) \\ \Leftrightarrow\ \ \forall x, y_0, y_1.\ \left(\begin{array}{l} \left(\begin{array}{c} x \longrightarrow_0 y_0 \\ \wedge\ \ x \longrightarrow_1 y_1 \end{array} \right) \\ \Rightarrow\ \ \exists z.\ \left(\begin{array}{c} y_0 \overset{*}{\longrightarrow}_1 z \\ \wedge\ \ y_1 \overset{*}{\longrightarrow}_0 z \end{array} \right) \end{array} \right) \end{array} \right)$$

$$(\mathsf{Wellf1}) \qquad \forall r : \mathsf{A} \to \mathsf{A} \to \mathsf{bool}.$$
$$\left(\begin{array}{l} \mathsf{Wellf}(r) \\ \Leftrightarrow\ \ \forall p : \mathsf{A} \to \mathsf{bool}.\ \left(\begin{array}{l} \exists x.\ p(x) \\ \Rightarrow\ \ \exists x.\ \left(\begin{array}{l} p(x) \\ \wedge\ \ \neg\exists y.\ \left(\begin{array}{c} p(y) \\ \wedge\ \ r(y, x) \end{array} \right) \end{array} \right) \end{array} \right) \end{array} \right)$$

Note that wellfoundedness and termination are no first-order properties.[24]

The transitivity lemma

(5) $u_0^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}^{\forall,\mathrm{w}} u_2^{\forall,\mathrm{w}},\ \neg u_0^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}^{\forall,\mathrm{w}} u_1^{\forall,\mathrm{w}},\ \neg u_1^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}^{\forall,\mathrm{w}} u_2^{\forall,\mathrm{w}},\ \neg\mathsf{Wellf}(\mathsf{Rev}(\longrightarrow^{\forall,\mathrm{w}}));\ u_0^{\forall,\mathrm{w}}$

can be shown by induction on $u_0^{\forall,\mathrm{w}}$ in $\longrightarrow^{\forall,\mathrm{w}}$. Note that we need the wellfoundedness because otherwise $\overset{*}{\longrightarrow}^{\forall,\mathrm{w}}$ may be a proper super-relation of the transitive closure of $\longrightarrow^{\forall,\mathrm{w}}$. I.e. the transitive closure is the smallest solution of $(*1)$ and in case of wellfoundedness there is only one single solution.

The following lemmas have very simple non-inductive proofs that expand the definition (Wellf1) twice. Note that the expansion of a logical equivalence is nothing but (γ-steps followed by) a kind of Rewrite-step because formulas can be seen as higher-order terms of type bool and the logical equivalence as the equality of type bool.

(6a) $\neg\mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}})),\ \mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}}))$

(6b) $\neg\mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}})),\ \mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_1^{\forall,\mathrm{w}}))$

Note that commutativity of Union implies that (6a) and (6b) are equivalent, but to prove $\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}} = \longrightarrow_1^{\forall,\mathrm{w}} \cup \longrightarrow_0^{\forall,\mathrm{w}}$ we need extensionality which we do not want to discuss here.

Now we are going to show the generalized Newman lemma, namely that wellfoundedness of the reverse of the union plus local commutation implies commutation.

$$\neg\mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}})),\ \neg\mathsf{LComm}(\longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}}),\ \mathsf{Comm}(\longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})$$

Expanding the definition (Comm1), three liberalized δ-steps, and two α-steps yield

$$\neg x^{\forall,\mathrm{s}} \overset{*}{\longrightarrow}_0^{\forall,\mathrm{w}} z_0^{\forall,\mathrm{s}},\ \neg x^{\forall,\mathrm{s}} \overset{*}{\longrightarrow}_1^{\forall,\mathrm{w}} z_1^{\forall,\mathrm{s}},\ \exists z.\left(\begin{array}{c} z_0^{\forall,\mathrm{s}} \overset{*}{\longrightarrow}_1^{\forall,\mathrm{w}} z \\ \wedge\ \ z_1^{\forall,\mathrm{s}} \overset{*}{\longrightarrow}_0^{\forall,\mathrm{w}} z \end{array}\right),$$
$$\neg\mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}})),\ \neg\mathsf{LComm}(\longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})$$

Now, since we want to do induction on $x^{\forall,\mathrm{s}}$, we start a new proof tree for

(7) $$\neg x^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}_0^{\forall,\mathrm{w}} z_0^{\forall,\mathrm{w}},\ \neg x^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}_1^{\forall,\mathrm{w}} z_1^{\forall,\mathrm{w}},\ \exists z.\left(\begin{array}{c} z_0^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}_1^{\forall,\mathrm{w}} z \\ \wedge\ \ z_1^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}_0^{\forall,\mathrm{w}} z \end{array}\right),$$
$$\neg\mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}})),\ \neg\mathsf{LComm}(\longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}});$$
$$x^{\forall,\mathrm{w}},\ <^{\exists}(x^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})$$

Note that this differs from the previous sequent (which can immediately be closed by lemma application of (7)) not only in that all free universal variables are weak now (which we also could have achieved by using non-liberalized δ-steps before instead of the liberalized ones) but also in that $x^{\forall,\mathrm{w}}$ is included in the weight, which is necessary for our intended induction. Actually we have set the weight directly to $x^{\forall,\mathrm{w}}$ for simplicity. Note that if the heuristic knowledge to recognize the above sequent as the likely induction hypothesis is not present, our calculi violate our design goal of a natural flow of information (cf. Section 2.1) because we sometime later realize that we should have started a new proof tree. With implemented calculi, however, this violation is no problem because one just has to implement a destructive inference rule that automatically splits a proof tree at a given position, reorganizes the former subtree into a new individual proof tree, and closes the cut branch by lemma or induction hypothesis application of the sequent of the new tree.

Moreover, we have added a free existential variable for the induction ordering

$$<^{\exists} : \mathsf{A} \to \mathsf{A} \to \mathsf{A} \to (\mathsf{A} \to \mathsf{A} \to \mathsf{bool}) \to (\mathsf{A} \to \mathsf{A} \to \mathsf{bool}) \to \mathsf{A} \to \mathsf{A} \to \mathsf{bool}$$

where the last two arguments will be supplied in infix notation below. Note that we have not supplied any induction quasi-ordering, but instead assume it to be the empty relation as in the discussion after Theor. 14.3, so that the sequents (5) and (6) can be omitted from the set M in Theor. 14.3.

We set our variable-condition $R := \{<^{\exists}\} \times \{x^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}}\}$ in order to have all weak free universal variables of (7) in the set Y of Theor. 14.3.

Expansion of the equivalence $(*1)$ in the first formula of (7), a β-, a liberalized δ- and an α-step yield:

$$(7.1)\;\; x^{\forall,\mathrm{w}} \neq z_0^{\forall,\mathrm{w}},\;\; \neg x^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z_1^{\forall,\mathrm{w}},\;\; \exists z. \left(\begin{array}{rl} & z_0^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z \\ \wedge & z_1^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z \end{array}\right),\;\; \ldots;\;\; \ldots$$

$$(7.2)\;\; \neg x^{\forall,\mathrm{w}} \longrightarrow_0^{\forall,\mathrm{w}} y_0^{\forall,\mathrm{s}},\;\; \neg y_0^{\forall,\mathrm{s}} \overset{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z_0^{\forall,\mathrm{w}},\;\; \neg x^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z_1^{\forall,\mathrm{w}},\;\; \exists z. \left(\begin{array}{rl} & z_0^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z \\ \wedge & z_1^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z \end{array}\right),$$
$$\neg\mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}})),\;\; \neg\mathsf{LComm}(\longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}});$$
$$x^{\forall,\mathrm{w}},\;\; <^{\exists}(x^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})$$

Rewriting with the first formula of (7.1) yields:

$$(7.1.1)\;\; \neg x^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z_1^{\forall,\mathrm{w}},\;\; \exists z. \left(\begin{array}{rl} & x^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z \\ \wedge & z_1^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z \end{array}\right),\;\; \ldots;\;\; \ldots$$

which is easily proved by setting z to $z_1^{\forall,\mathrm{w}}$ in a γ-step. Expansion of the equivalence $(*1)$ in the third formula of (7.2), a β-, a liberalized δ- and an α-step yield:

$$(7.2.1)\;\; x^{\forall,\mathrm{w}} \neq z_1^{\forall,\mathrm{w}},\;\; \neg x^{\forall,\mathrm{w}} \longrightarrow_0^{\forall,\mathrm{w}} y_0^{\forall,\mathrm{s}},\;\; \neg y_0^{\forall,\mathrm{s}} \overset{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z_0^{\forall,\mathrm{w}},\;\; \exists z. \left(\begin{array}{rl} & z_0^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z \\ \wedge & z_1^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z \end{array}\right),\;\; \ldots;\;\; \ldots$$

$$(7.2.2)\;\; \neg x^{\forall,\mathrm{w}} \longrightarrow_1^{\forall,\mathrm{w}} y_1^{\forall,\mathrm{s}},\;\; \neg y_1^{\forall,\mathrm{s}} \overset{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z_1^{\forall,\mathrm{w}},\;\; \neg x^{\forall,\mathrm{w}} \longrightarrow_0^{\forall,\mathrm{w}} y_0^{\forall,\mathrm{s}},\;\; \neg y_0^{\forall,\mathrm{s}} \overset{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z_0^{\forall,\mathrm{w}},$$
$$\exists z. \left(\begin{array}{rl} & z_0^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z \\ \wedge & z_1^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z \end{array}\right),\;\; \neg\mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}})),\;\; \neg\mathsf{LComm}(\longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}});$$
$$x^{\forall,\mathrm{w}},\;\; <^{\exists}(x^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})$$

Rewriting with the first formula of (7.2.1) yields:

$$(7.2.1.1)\;\; \neg x^{\forall,\mathrm{w}} \longrightarrow_0^{\forall,\mathrm{w}} y_0^{\forall,\mathrm{s}},\;\; \neg y_0^{\forall,\mathrm{s}} \overset{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z_0^{\forall,\mathrm{w}},\;\; \exists z. \left(\begin{array}{rl} & z_0^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z \\ \wedge & x^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z \end{array}\right),\;\; \ldots;\;\; \ldots$$

Now we have to regenerate the literal $\neg x^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z_0^{\forall,\mathrm{w}}$ (which a tableau proof would still have available from (7)) by application of $(*1)$ and then close this subtree by setting z to $z_0^{\forall,\mathrm{w}}$.

Expansion of (LComm1) in (7.2.2), γ-, β- and liberalized δ-steps yield two tautologies plus

$$\begin{array}{l}
(7.2.2.1)\ \neg y_0^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} y_2^{\forall,\mathrm{s}},\ \ \neg y_1^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} y_2^{\forall,\mathrm{s}},\\
\qquad \neg x^{\forall,\mathrm{w}} \longrightarrow_1^{\forall,\mathrm{w}} y_1^{\forall,\mathrm{s}},\ \ \neg y_1^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z_1^{\forall,\mathrm{w}},\ \ \neg x^{\forall,\mathrm{w}} \longrightarrow_0^{\forall,\mathrm{w}} y_0^{\forall,\mathrm{s}},\ \ \neg y_0^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z_0^{\forall,\mathrm{w}},\\
\qquad \exists z. \left(\begin{array}{ll} & z_0^{\forall,\mathrm{w}} \stackrel{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z \\ \wedge & z_1^{\forall,\mathrm{w}} \stackrel{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z \end{array}\right),\ \ \neg\mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}})),\ \ \neg\mathsf{LComm}(\longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}});\\
\hfill x^{\forall,\mathrm{w}},\ \ <^{\exists}(x^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})
\end{array}$$

Applying (7) as an induction hypothesis according to Theor. 14.3 with substitution

$$\{x^{\forall,\mathrm{w}} \mapsto y_0^{\forall,\mathrm{s}},\ \ z_1^{\forall,\mathrm{w}} \mapsto y_2^{\forall,\mathrm{s}}\}$$

yields four tautologies and

$$\begin{array}{l}
(7.2.2.1.1)\ \neg z_0^{\forall,\mathrm{w}} \stackrel{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} y_3^{\forall,\mathrm{s}},\ \ \neg y_2^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} y_3^{\forall,\mathrm{s}},\ \ \neg y_0^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} y_2^{\forall,\mathrm{s}},\ \ \neg y_1^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} y_2^{\forall,\mathrm{s}},\\
\qquad \neg x^{\forall,\mathrm{w}} \longrightarrow_1^{\forall,\mathrm{w}} y_1^{\forall,\mathrm{s}},\ \ \neg y_1^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z_1^{\forall,\mathrm{w}},\ \ \neg x^{\forall,\mathrm{w}} \longrightarrow_0^{\forall,\mathrm{w}} y_0^{\forall,\mathrm{s}},\ \ \neg y_0^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z_0^{\forall,\mathrm{w}},\\
\qquad \exists z. \left(\begin{array}{ll} & z_0^{\forall,\mathrm{w}} \stackrel{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z \\ \wedge & z_1^{\forall,\mathrm{w}} \stackrel{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z \end{array}\right),\ \ \neg\mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}})),\ \ \neg\mathsf{LComm}(\longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}});\\
\hfill x^{\forall,\mathrm{w}},\ \ <^{\exists}(x^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})
\end{array}$$

$$(7.2.2.1.2)\ y_0^{\forall,\mathrm{s}} <^{\exists}(x^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})\ x^{\forall,\mathrm{w}},\ \ \ldots,\ \ \neg x^{\forall,\mathrm{w}} \longrightarrow_0^{\forall,\mathrm{w}} y_0^{\forall,\mathrm{s}},\ \ \ldots$$

$$\begin{array}{l}
(7.2.2.1.3)\ \forall p : \mathsf{A} \rightarrow \mathsf{bool}.\\
\quad \left(\exists x.\ p(x) \Rightarrow \exists x. \left(\begin{array}{ll} & p(x) \\ \wedge & \neg\exists y. \left(\begin{array}{ll} & p(y) \\ \wedge & y <^{\exists}(x^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})\ x \end{array}\right)\end{array}\right)\right),\\
\hfill \ldots,\ \ \neg\mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}})),\ \ \ldots
\end{array}$$

$$(7.2.2.1.4)\ \forall x, y : \mathsf{A}. \left(\begin{array}{ll} & x <^{\exists}(x^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})\ y \\ \Leftrightarrow & x <^{\exists}(y_0^{\forall,\mathrm{s}}, z_0^{\forall,\mathrm{w}}, y_2^{\forall,\mathrm{s}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})\ y \end{array}\right),\ \ \ldots$$

where (7.2.2.1.1) is presented after application of a liberalized δ- and an α-step. The situation of the first two lines of (7.2.2.1.1) (seen as an antecedent) can be depicted as follows:

$$\begin{tikzcd}
x^{\forall,\mathrm{w}} \arrow[r, "0"'] \arrow[d, "1"'] & y_0^{\forall,\mathrm{s}} \arrow[r, "*", "0"'] \arrow[d, "1"', "*"] & z_0^{\forall,\mathrm{w}} \arrow[d, "1"', "*"] \\
y_1^{\forall,\mathrm{s}} \arrow[r, "*", "0"'] \arrow[d, "1"', "*"] & y_2^{\forall,\mathrm{s}} \arrow[r, "*", "0"'] & y_3^{\forall,\mathrm{s}} \\
z_1^{\forall,\mathrm{w}} & &
\end{tikzcd}$$

Application of (5) as a lemma yields (besides a sequent that can be closed by lemma application of (6a))

$$\begin{array}{l}
(7.2.2.1.1.1)\ \neg y_1^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} y_3^{\forall,\mathrm{s}},\ \ \neg z_0^{\forall,\mathrm{w}} \stackrel{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} y_3^{\forall,\mathrm{s}},\ \ \neg y_2^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} y_3^{\forall,\mathrm{s}},\ \ \neg y_0^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} y_2^{\forall,\mathrm{s}},\ \ \neg y_1^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} y_2^{\forall,\mathrm{s}},\\
\qquad \neg x^{\forall,\mathrm{w}} \longrightarrow_1^{\forall,\mathrm{w}} y_1^{\forall,\mathrm{s}},\ \ \neg y_1^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z_1^{\forall,\mathrm{w}},\ \ \neg x^{\forall,\mathrm{w}} \longrightarrow_0^{\forall,\mathrm{w}} y_0^{\forall,\mathrm{s}},\ \ \neg y_0^{\forall,\mathrm{s}} \stackrel{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z_0^{\forall,\mathrm{w}},\\
\qquad \exists z. \left(\begin{array}{ll} & z_0^{\forall,\mathrm{w}} \stackrel{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}} z \\ \wedge & z_1^{\forall,\mathrm{w}} \stackrel{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}} z \end{array}\right),\ \ \neg\mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}})),\ \ \neg\mathsf{LComm}(\longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}});\\
\hfill x^{\forall,\mathrm{w}},\ \ <^{\exists}(x^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})
\end{array}$$

Applying (7) as an induction hypothesis with substitution $\{x^{\forall,\mathrm{w}}\mapsto y_1^{\forall,\mathrm{s}},\ z_0^{\forall,\mathrm{w}}\mapsto y_3^{\forall,\mathrm{s}}\}$ yields four tautologies and

(7.2.2.1.1.1.1) $\neg z_0^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}}\, y_4^{\forall,\mathrm{s}},\ \ \neg z_1^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}}\, y_4^{\forall,\mathrm{s}},\ \ \neg y_3^{\forall,\mathrm{s}} \overset{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}}\, y_4^{\forall,\mathrm{s}},$

$$\neg y_1^{\forall,\mathrm{s}} \overset{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}}\, y_3^{\forall,\mathrm{s}},\ \ \neg z_0^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}}\, y_3^{\forall,\mathrm{s}},\ \ \ldots,\ \ \exists z.\left(\begin{array}{ll} & z_0^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_1^{\forall,\mathrm{w}}\, z \\ \wedge & z_1^{\forall,\mathrm{w}} \overset{*}{\longrightarrow}{}_0^{\forall,\mathrm{w}}\, z \end{array}\right),\ \ \ldots$$

(7.2.2.1.1.1.2) $y_1^{\forall,\mathrm{s}} <^{\exists}(x^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})\ x^{\forall,\mathrm{w}},\ \ \ldots,\ \ \neg x^{\forall,\mathrm{w}} \longrightarrow_1^{\forall,\mathrm{w}} y_1^{\forall,\mathrm{s}},\ \ \ldots$

(7.2.2.1.1.1.3) $\forall p : \mathsf{A} \to \mathsf{bool}.$

$$\left(\exists x.\ p(x) \Rightarrow \exists x.\left(\begin{array}{ll} & p(x) \\ \wedge & \neg\exists y.\left(\begin{array}{ll} & p(y) \\ \wedge & y <^{\exists}(x^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})\ x \end{array}\right)\end{array}\right)\right),$$

$$\ldots,\ \ \neg\mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}})),\ \ \ldots$$

(7.2.2.1.1.1.4) $\forall x, y : \mathsf{A}.\left(\begin{array}{ll} & x <^{\exists}(x^{\forall,\mathrm{w}}, z_0^{\forall,\mathrm{w}}, z_1^{\forall,\mathrm{w}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})\ y \\ \Leftrightarrow & x <^{\exists}(y_1^{\forall,\mathrm{s}}, y_3^{\forall,\mathrm{s}}, z_1^{\forall,\mathrm{w}}, \longrightarrow_0^{\forall,\mathrm{w}}, \longrightarrow_1^{\forall,\mathrm{w}})\ y \end{array}\right),\ \ \ldots$

where (7.2.2.1.1.1.1) is presented after application of a liberalized δ- and an α-step, which can be depicted as

$$\begin{tikzcd}
x^{\forall,\mathrm{w}} \arrow[r, "0"'] \arrow[d, "1"'] & y_0^{\forall,\mathrm{s}} \arrow[r, "*", "0"'] & z_0^{\forall,\mathrm{w}} \arrow[d, "1"', "*"] \\
y_1^{\forall,\mathrm{s}} \arrow[rr, "*", "0"'] \arrow[d, "1"', "*"] & & y_3^{\forall,\mathrm{s}} \arrow[d, "1"', "*"] \\
z_1^{\forall,\mathrm{w}} \arrow[rr, "*", "0"'] & & y_4^{\forall,\mathrm{s}}
\end{tikzcd}$$

and after a lemma application of (5) (producing another goal closed by lemma application of (6b)).

Now (7.2.2.1.1.1.1) can be closed after setting z to $y_4^{\forall,\mathrm{s}}$ in a γ-step. When we finally apply the existential R-substitution $\{<^{\exists} \mapsto \lambda v_0, \ldots, v_4.\ (\mathsf{Rev}(v_3 \cup v_4))\}$ and $\lambda\beta$-reduce, we get the following open goals:

(7.2.2.1.2') $\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}}, y_0^{\forall,\mathrm{s}}, x^{\forall,\mathrm{w}}),\ \ \ldots,\ \ \neg x^{\forall,\mathrm{w}} \longrightarrow_0^{\forall,\mathrm{w}} y_0^{\forall,\mathrm{s}},\ \ \ldots$

(7.2.2.1[.1.1].3') $\forall p : \mathsf{A} \to \mathsf{bool}.$

$$\left(\exists x.\ p(x) \Rightarrow \exists x.\left(\begin{array}{ll} & p(x) \\ \wedge & \neg\exists y.\left(\begin{array}{ll} & p(y) \\ \wedge & \mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}}, y, x) \end{array}\right)\end{array}\right)\right),$$

$$\ldots,\ \ \neg\mathsf{Wellf}(\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}})),\ \ \ldots$$

(7.2.2.1[.1.1].4') $\forall x, y : \mathsf{A}.\left(\begin{array}{ll} & \mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}}, x, y) \\ \Leftrightarrow & \mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}}, x, y) \end{array}\right),\ \ \ldots$

(7.2.2.1.1.1.2') $\mathsf{Rev}(\longrightarrow_0^{\forall,\mathrm{w}} \cup \longrightarrow_1^{\forall,\mathrm{w}}, y_1^{\forall,\mathrm{s}}, x^{\forall,\mathrm{w}}),\ \ \ldots,\ \ \neg x^{\forall,\mathrm{w}} \longrightarrow_1^{\forall,\mathrm{w}} y_1^{\forall,\mathrm{s}},\ \ \ldots$

which can be easily closed.

20 Optimizations

Note that (as far as Theor. 14.1 and Theor. 14.3 are concerned) the choice-conditions do not have any influence on our proofs when we never instantiate strong free universal variables and when, in the liberalized δ-steps, we always choose a completely new strong free universal variable $x^{\forall,\mathrm{s}}$ that does not occur elsewhere. Thus, the choice-conditions may be omitted in an implementation. We could, however, use them for the following purposes:

1. We could use the choice-conditions in order to weaken our requirements for our set of axioms $\mathcal{AX}$: Instead of (weak) $\mathrm{V}_{\exists}{\times}\mathrm{V}_{\forall}$-validity of $\mathcal{AX}$, (C, R)-strong validity of $\mathcal{AX}$ (which is logically weaker, cf. Lemma 10.2) is sufficient for Theor. 13.6.

2. We can simulate the behavior of an improved version of the δ^{++}-rule of Beckert &al. (1993) by equating different strong free universal variables whose C-values are initially equal or have become logically equivalent during the proof. Note that this does not anymore require a functional and extensional behavior of choice-conditions as in Wirth (1998). There we had to require that, for $(x^{\forall,\mathrm{s}}, A) \in C$, the value for $x^{\forall,\mathrm{s}}$ is not just an arbitrary one from the set of values that make A valid, but a unique element of this set given by some choice-function. In the present version (due to the changed notion of strong validity) it is possible to globally replace not only a free existential variable, but also a strong free universal variable $x^{\forall,\mathrm{s}}$ with any term that (if possible) makes $C(x^{\forall,\mathrm{s}})$ true.

   Expressed with Hilbert's ε-terms (as indicated in Section 9), our treatment is similar to a structure sharing version of the merely intensional treatment of ε-terms in Giese & Ahrendt (1999). Note that our choice-conditions even do not imply a functional dependence of $\epsilon(\pi)(\tau)(y^{\forall,\mathrm{s}})$ from $C(y^{\forall,\mathrm{s}})$; instead the choice of a special value is a step in a proof similar to the instantiation of a free existential variable, and we do not have to commit to this choice for other occurrences of the same ε-term. This means that our choice-conditions work like the word "some" in the in the English language. E.g., "Some human loves some human" is like $\mathsf{Loves}(x^{\forall,\mathrm{s}}, y^{\forall,\mathrm{s}})$ with $C(x^{\forall,\mathrm{s}}) = \mathsf{Human}(x^{\forall,\mathrm{s}})$ and $C(y^{\forall,\mathrm{s}}) = \mathsf{Human}(y^{\forall,\mathrm{s}})$, or like

   $$\mathsf{Loves}(\varepsilon x.\ \mathsf{Human}(x),\ \varepsilon x.\ \mathsf{Human}(x))$$

   and follows from $\mathsf{Loves}(\mathsf{Jack}, \mathsf{Jill})$, $\mathsf{Human}(\mathsf{Jack})$, $\mathsf{Human}(\mathsf{Jill})$. There is more on this subject in Wirth (2002).

3. Moreover, the choice-conditions may be used to get more interesting solutions:

Example 20.1
Starting with the empty proof forest and hypothesizing

$$\forall x.\ \mathsf{Q}(x,x),\quad \exists y.\ \big(\ \neg\mathsf{Q}(y,y) \wedge \neg\mathsf{P}(y)\ \big),\quad \mathsf{P}(z^{\exists})$$

with the rules at the end of Section 2.3 we can produce a proof tree with the leaves

$$\neg\mathsf{Q}(y^{\exists},y^{\exists}),\quad \mathsf{Q}(x^{\forall,\mathrm{s}},x^{\forall,\mathrm{s}}),\quad \exists y.\ \big(\ \neg\mathsf{Q}(y,y) \wedge \neg\mathsf{P}(y)\ \big),\quad \mathsf{P}(z^{\exists})$$

and

$$\neg\mathsf{P}(y^{\exists}),\quad \mathsf{Q}(x^{\forall,\mathrm{s}},x^{\forall,\mathrm{s}}),\quad \exists y.\ \big(\ \neg\mathsf{Q}(y,y) \wedge \neg\mathsf{P}(y)\ \big),\quad \mathsf{P}(z^{\exists})$$

and the $\emptyset$-choice-condition $\{(x^{\forall,\mathrm{s}}, \neg\mathsf{Q}(x^{\forall,\mathrm{s}},x^{\forall,\mathrm{s}}))\}$.

The existential $\emptyset$-substitution $\{y^{\exists}\mapsto x^{\forall,\mathrm{s}},\ z^{\exists}\mapsto x^{\forall,\mathrm{s}}\}$ closes the proof tree via an Instantiation step. The solution $x^{\forall,\mathrm{s}}$ for our query variable $z^{\exists}$ is not very interesting unless the choice-condition tells us to choose $x^{\forall,\mathrm{s}}$ in such a way that $\mathsf{Q}(x^{\forall,\mathrm{s}},x^{\forall,\mathrm{s}})$ becomes false.

Note that if we had applied the δ-rule instead of the liberalized δ-rule in the above proof, i.e. if we had introduced $x^{\forall,\mathrm{w}}$ instead of $x^{\forall,\mathrm{s}}$, then we would not only be unable to provide any information on our query variable (because the choice-condition is empty), but we would even be unable to finish our proof because due to the new variable-condition $R = \{(z^{\exists}, x^{\forall,\mathrm{w}})\}$ we cannot apply $\{y^{\exists}\mapsto x^{\forall,\mathrm{w}},\ z^{\exists}\mapsto x^{\forall,\mathrm{w}}\}$ anymore, because it is not an existential R-substitution. With the (weak) δ-rule, all we can show instead is

$$\forall x.\ \mathsf{Q}(x,x),\quad \exists y.\ \big(\ \neg\mathsf{Q}(y,y) \wedge \neg\mathsf{P}(y)\ \big),\quad \exists z.\ \mathsf{P}(z)$$

Thus, it is obvious that the liberalized δ-rule is not only superior[25] to the (non-liberalized) δ-rule w.r.t. theorem proving but also w.r.t. computation of answers and solutions. Nevertheless, when interested only in proving theorems not containing strong free universal variables, the choice-conditions do not produce any overhead because they can simply be omitted; thereby leaving the strong free universal variables unspecified just like the Skolem functions in Skolemizing deduction.

The only overhead compared to the standard framework of Skolemization seems to be that we have to compute transitive closures when checking whether a substitution σ is really an existential R-substitution and when computing the σ-update of R. But we actually do not have to compute the transitive closure at all, because the only essential thing is the circularity-check which can be done on a graph generating the transitive closures. This checking is in the worst case linear in

$$|R| + \sum_{\sigma} \big(\ |U_\sigma| + |E_\sigma|\ \big)$$

and is expected to perform at least as well as an optimally integrated version (i.e. one without conversion of term-representation) of the linear unification algorithm of Paterson & Wegman (1978) in the standard framework of Skolemization and unification. (Of course, the checking for existential R-substitutions can also be implemented with any other unification algorithm.)

20.1 Variable-Condition versus Free Universal Variables

Not really computing the transitive closure enables another refinement that allows us to go even beyond the fascinating *strong Skolemization* of Nonnengart (1996). The basic idea of Nonnengart (1996) can be translated into our framework in the following simplified way.

Instead of proving $\forall x.\ (A{\vee}B)$ it may be advantageous to prove the stronger $\forall x.\ A \vee \forall x.\ B$, because after applications of α- and liberalized δ-rules to $\forall x.\ A \vee \forall x.\ B$, resulting in $A\{x{\mapsto}x_A^{\forall,\mathrm{s}}\}$, $B\{x{\mapsto}x_B^{\forall,\mathrm{s}}\}$, the variable-conditions introduced for $x_A^{\forall,\mathrm{s}}$ and $x_B^{\forall,\mathrm{s}}$ may be smaller than the variable-condition introduced for $y^{\forall,\mathrm{s}}$ after applying these rules to $\forall x.\ (A{\vee}B)$, resulting in $A\{x{\mapsto}y^{\forall,\mathrm{s}}\}$, $B\{x{\mapsto}y^{\forall,\mathrm{s}}\}$, i.e. $\mathcal{V}_{\mathrm{free}}(A)$ and $\mathcal{V}_{\mathrm{free}}(B)$ may be *proper* subsets of $\mathcal{V}_{\mathrm{free}}(A,B)$. Therefore the proof of $\forall x.\ A \vee \forall x.\ B$ may be simpler than the proof of $\forall x.\ (A{\vee}B)$. The nice aspect of strong Skolemization roughly translated into our framework is that the intermediately sized $\mathcal{V}_{\mathrm{free}}(A) \times \{x_A^{\forall,\mathrm{s}}\} \ \cup\ \mathcal{V}_{\mathrm{free}}(A,B) \times \{x_B^{\forall,\mathrm{s}}\}$ is added to the variable-condition, but only a single Skolem function f is introduced with $x_A^{\forall,\mathrm{s}}$ represented as $f(A',X)$ and $x_B^{\forall,\mathrm{s}}$ represented as $f(A',B' \backslash A')$ where X are some new free existential variables, $A' := \mathrm{V}_\exists \cap R^*\langle\mathcal{V}_{\mathrm{free}}(A)\rangle$, and $B' := \mathrm{V}_\exists \cap R^*\langle\mathcal{V}_{\mathrm{free}}(B)\rangle$. Thus, $x_B^{\forall,\mathrm{s}}$ still becomes dependent on the free variables of the whole disjunction, so that—due to this asymmetry[26]—it may make an important difference to prove $\forall x.\ (A{\vee}B)$ or $\forall x.\ (B{\vee}A)$.

Now, if we do not really compute the transitive closures in our strong version, we can prove $A\{x{\mapsto}x_A^{\forall,\mathrm{s}}\}$, $B\{x{\mapsto}x_B^{\forall,\mathrm{s}}\}$ in parallel and may later decide to prove the stronger $A\{x{\mapsto}y^{\forall,\mathrm{s}}\}$, $B\{x{\mapsto}y^{\forall,\mathrm{s}}\}$ instead, simply by merging the nodes for $x_A^{\forall,\mathrm{s}}$ and $x_B^{\forall,\mathrm{s}}$ and substituting $x_A^{\forall,\mathrm{s}}$ and $x_B^{\forall,\mathrm{s}}$ with $y^{\forall,\mathrm{s}}$. Of course, we have to pay the price of checking whether all substitutions are still existential R-substitutions.

Finally note that the same conflict and solution apply to

$$\forall x.\ (A{\wedge}B) \quad \text{vs.} \quad \forall x.\ A \wedge \forall x.\ B,$$

although these formulas are logically equivalent: The latter in general produces smaller variable-conditions (unless $\mathcal{V}_{\mathrm{free}}(A) = \mathcal{V}_{\mathrm{free}}(B)$) but the former less free universal variables (Skolem functions) and each of the two may reduce the proof size.

20.2 Improving Multiple γ-Rule Applications

Another optimization, inspired by the ideas of Section 7 of Giese (1998) and Appendix B of Wirth (1997), improves the behavior of multiple γ-rule applications to the same formula. It requires a new kind of free existential variables which are not used for direct instantiation but as generators for the usual kind of free existential variables. In the tableau community these variables are sometimes called "universal" (cf. e.g. Beckert & Hähnle (1998)), but they have nothing to do with our free universal variables here. Thus, we call them generator variables and denote them with $x^{\exists,\mathrm{g}}$ and $\mathrm{V}_{\exists,\mathrm{g}}$, &c.. Instead of the γ-rule say

$$\frac{\varGamma \quad \exists x.\ A \quad \varPi}{A\{x{\mapsto}x^{\exists}\} \quad \varGamma \quad \exists x.\ A \quad \varPi}$$

we take a rule like

$$\frac{\varGamma \quad \exists x.\ A \quad \varPi}{A\{x{\mapsto}x^{\exists,\mathrm{g}}\} \quad \varGamma \quad \varPi}$$

where $\exists x.\ A$ is removed and $x^{\exists,\mathrm{g}}$ is a new generator variable. Other γ-rules are changed analogously. α-rules are not changed and the β-rules at the end of Section 2.3 are restricted in their applicability by the restriction of $\mathcal{V}_{\exists,\mathrm{g}}(A) \cap \mathcal{V}_{\exists,\mathrm{g}}(B) = \emptyset$, which (together with the condition that generator variables do not occur in root sequents of Hypothesizing steps and substitutions of Instantiation steps) guarantees that for each generator variable there is always a single branch in a tree that contains all occurrences of this generator variable. To enable blocked β-rule applications and for Instantiation, we need a *generation* rule like

$$\frac{\Gamma \quad A \quad \Pi}{A\{x^{\exists,\mathrm{g}} \mapsto x^{\exists}\} \quad \Gamma \quad A \quad \Pi}$$

The δ-rules at the end of Section 2.3 either get restricted by $\mathcal{V}_{\exists,\mathrm{g}}(A) = \emptyset$ or otherwise we can proceed in the following less simple but more powerful way: We must treat generator variables like free existential variables and the generation rule must replace each strong (weak ones analogously) free universal variable $y^{\forall,\mathrm{s}}$ from $\mathcal{V}_{\forall,\mathrm{s}}(A) \cap \langle\{x^{\exists,\mathrm{g}}\}\rangle R^{+}$ in $A\{x^{\exists,\mathrm{g}} \mapsto x^{\exists}\}$ with a new one, say $y'^{\forall,\mathrm{s}}$, and add to the variable-condition R a copy of the graph of $\langle\{x^{\exists,\mathrm{g}}\}\rangle R^{*}$ with $x^{\exists}$ instead of $x^{\exists,\mathrm{g}}$ and $y'^{\forall,\mathrm{s}}$ instead of $y^{\forall,\mathrm{s}}$ &c., and add to the choice-condition C something like $(y'^{\forall,\mathrm{s}}, (C(y^{\forall,\mathrm{s}}))\{x^{\exists,\mathrm{g}} \mapsto x^{\exists},\ \ y^{\forall,\mathrm{s}} \mapsto y'^{\forall,\mathrm{s}},\ \ \ldots\})$. The nice treatment in Section 7 of Giese (1998) makes the reason for this seemingly complicated procedure obvious by means of Hilbert's ε-terms.

The crucial step in order to include this into our framework here is to change the notion of $(\tau, e, \mathcal{A})$-validity such that a generator variable $x^{\exists,\mathrm{g}}$ is treated like a free existential variable with the exception that its value may also be chosen from the values of the free existential variables that have been generated from it; i.e. a value of $x^{\exists,\mathrm{g}}$ establishing the validity must exist among $\epsilon(e)(\tau)(x^{\exists,\mathrm{g}})$ and the $\epsilon(e)(\tau)(x^{\exists})$ for the free existential variables $x^{\exists}$ generated from $x^{\exists,\mathrm{g}}$. Without this, the generation rule would not preserve solutions.

Now, if the A in the γ-rule is a literal or a blocked β-formula, then the new γ-rule plus n generation steps have the effect of n applications of the old γ-rule and no improvement takes place. Otherwise, however, several inference rules may be applied after the new γ-rule, and when we suddenly discover that we need say $\mathsf{P}(x^{\exists,\mathrm{g}})$ twice, then we can apply two generation steps instead of repeating the whole subtree up to the γ-rule application.

All in all, we have to admit that the possibilities to improve multiple γ-rule applications are poor in sequent and tableau calculi. In a matrix representation like in Wallen (1990), however, it is possible to dynamically increase the multiplicity and to let all existential variables be generating, no matter whether all occurrences of each variable are on the same branch or not. Thus, an implementation should use matrix calculi instead of the presentationally simpler sequent and tableau calculi used in this paper because then the β- and δ-rules do not suffer from the severe restrictions explained above.

21 Conclusion

After introducing the required interdisciplinary background, we have presented a combination of the following features in deductive theorem proving: raising, explicit variable dependency representation, preservation of solutions, and the liberalized δ-rule. To our knowledge[27] we have presented on the one hand the first sound combination of explicit variable dependency representation and the liberalized δ-rule. And on the other hand the first framework for preservation of solutions in full first-order logic. The difficulties described in Section 8 with the combination of the preservation of solutions and the liberalized δ-rule reveal unexpected details on the nature of the liberalized δ-rule.

The original motivation for our work, however, was not to combine these seemingly contrary features, but to provide the foundation for ITP, where the preservation of solutions is indispensable for the possibility to model *descente infinie*.

In the present version, lemma application and induction hypothesis application are included for the first time in all calculi: Wirth (1999) included only hypothesis application for the weak version of the calculi of Wirth (1998). For the strong version we surprisingly had to change the notion of strong validity in order to make lemma or induction hypothesis application possible. With this exception and besides lots of minor improvements, the calculi of this paper are the ones of the strong version of Wirth (1998), although we have omitted the word "strong" in the names of the notions wherever we did not present any weak version in this paper.

We have shown how to integrate *descente infinie* into state-of-the-art classical sequent and tableau calculi. The following aspects are novel compared to the concrete induction calculus of Wirth (1997): The tableau presentation, the possibility to use sequents of full first-order and higher-order formulas instead of literals only, and the important addition of free existential variables, i.e. the "dummies" of Prawitz (1960), making the major difference between the free variable calculi of Fitting (1996) and the calculi of Smullyan (1968). Contrary to Baaz &al. (1997) we really integrate *descente infinie*: When we start an inductive proof we do not restrict the applicable induction hypotheses. We can do mutual induction and invent completely new induction hypotheses, which can be sequents instead of literals only. Moreover, we can also generate induction hypotheses eagerly in the style of explicit induction, which enables goal-directedness w.r.t. induction hypotheses. Finally, we can have variable induction orderings. Thus, our calculi are much "fatter" than the "lean" calculus of Baaz &al. (1997) where all this is not possible. Furthermore, in Section 17 we exemplified that although tableau calculi may save repetition of formulas, sequent calculi have substantial advantages.

We hope that our examples illustrate that our modeling of *descente infinie* in sequent and tableau calculi can meet our design goals of Section 2.1 and is of practical relevance. A more convincing example that shows the features in their combined power requires an implementation with a graphical user interface for development as well as for presentation.

Acknowledgments: I would like to thank Paul Howard for a short e-mail communication on the Principle of Dependent Choice that was very helpful to me and Ulrich Kühler for his fascinating XQuodLibet system.

A Additional Lemmas

The following lemma—roughly speaking—only says that the Substitution-Lemma can be lifted to existential valuations as expected.

Lemma A.1 *Let R be a variable-condition and σ an R-substitution.*

1. *Let R' be a variable-condition with $R{\subseteq}R'$. Now:*
   Each existential $(\mathcal{A}, R')$-valuation is also an existential $(\mathcal{A}, R)$-valuation.

2. *Let R' the σ-update of R. For each existential $(\mathcal{A}, R')$-valuation e' there is some existential $(\mathcal{A}, R)$-valuation e s.t.*
$$S_e \quad = \quad S_{e'} \circ ({}_{\mathrm{V}_\exists \setminus \mathrm{dom}(\sigma)}{\upharpoonleft}\mathrm{id} \cup E_\sigma{\upharpoonright}_{\mathrm{V}_\exists}) \quad \cup \quad U_\sigma{\upharpoonright}_{\mathrm{V}_\exists}$$
   and for all $\delta \in \mathrm{V}_\forall \to \mathcal{A}$:
$$\epsilon(e)(\delta) \quad = \quad ({}_{\mathrm{V}_\exists \setminus \mathrm{dom}(\sigma)}{\upharpoonleft}\mathrm{id} \cup {}_{\mathrm{V}_\exists}{\upharpoonleft}\sigma) \;\circ\; \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta) \uplus \delta).$$

3. *Let (C', R') the extended σ-update of (C, R). For each existential $(\mathcal{A}, R')$-valuation e' and each π' that is $(e', \mathcal{A})$-compatible with (C', R'), there is some existential $(\mathcal{A}, R)$-valuation e s.t.*
$$S_e \quad = \quad (S_{\pi'} \cup {}_{\mathrm{V}_{\forall,\mathrm{w}}}{\upharpoonleft}\mathrm{id}) \;\circ\; (S_{e'} \circ ({}_{\mathrm{V}_\exists \setminus \mathrm{dom}(\sigma)}{\upharpoonleft}\mathrm{id} \cup E_\sigma{\upharpoonright}_{\mathrm{V}_\exists}) \cup U_\sigma{\upharpoonright}_{\mathrm{V}_\exists})$$
   and for all $\delta \in \mathrm{V}_\forall \to \mathcal{A}$, when τ abbreviates ${}_{\mathrm{V}_{\forall,\mathrm{w}}}{\upharpoonleft}\delta$:
$$\epsilon(e)(\delta) \quad = \quad ({}_{\mathrm{V}_\exists \setminus \mathrm{dom}(\sigma)}{\upharpoonleft}\mathrm{id} \cup {}_{\mathrm{V}_\exists}{\upharpoonleft}\sigma) \;\circ\; \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\epsilon(\pi')(\tau) \uplus \tau) \uplus \epsilon(\pi')(\tau) \uplus \tau),$$
   Note that $R \cup S_e \cup (R' \cup S_{e'} \cup S_{\pi'})^+{\upharpoonright}_{\mathrm{V}_\forall}$ is wellfounded here.

For dealing with quasi-existential R-substitutions semantically, we need the following technical sub-lemma to Lemma 11.2(5) and Lemma 12.3(5), which was removed from the end of Section 7. Note that, considering those variables that are constrained by the choice-condition C and replaced by the substitution σ, on the one hand, the set O contains the variables whose replacements are supported by the lemmas $\langle O \rangle Q_{C,\sigma}$ (cf. Definition 9.4). On the other hand, the set N contains the variables that are not supported by such lemmas, plus all the variables that are constrained by C and suffer from this missing support in the sense that they depend on these variables via the variable-condition R.

Lemma A.2 *Let C be an R-choice-condition, $\mathcal{A}$ a Σ-structure, σ a quasi-existential R-substitution. Let (C', R') be the extended σ-update of (C, R). Assume that we have O and N with $O \subseteq \mathrm{dom}(C) \cap \mathrm{dom}(\sigma) \subseteq O \uplus N$, $N \subseteq \mathrm{dom}(C) \setminus O$, $\mathrm{dom}(C) \cap \langle N \rangle R^+ \subseteq N$. Now, for any existential $(\mathcal{A}, R')$-valuation e' and any π' that is $(e', \mathcal{A})$-compatible with (C', R') s.t. $\langle O \rangle Q_{C,\sigma}$ is strongly $(\pi', e', \mathcal{A})$-valid, there are an existential $(\mathcal{A}, R)$-valuation e and a π that is $(e, \mathcal{A})$-compatible with (C, R) for which the following holds:*

1. *For any term or formula B (possibly containing some unbound variables from a set $W \subseteq \mathrm{V}_{\mathrm{bound}}$) with $N \cap \mathcal{V}(B) = \emptyset$, and for any $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$ and $\chi \in W \to \mathcal{A}$, when we set $\delta' := \epsilon(\pi')(\tau) \uplus \tau$ and $\delta := \epsilon(\pi)(\tau) \uplus \tau$:*
$$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \delta' \uplus \chi)(B\sigma) = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta \uplus \chi)(B).$$

2. *For any set of sequents G with $N \cap \mathcal{V}(G) = \emptyset$:*
$$G\sigma \textit{ is strongly } (\pi', e', \mathcal{A})\textit{-valid iff } G \textit{ is strongly } (\pi, e, \mathcal{A})\textit{-valid.}$$

B Proofs

Proof of Lemma 3.1
Since R^+ is clearly transitive, it suffices to show that it is wellfounded, because then it is irreflexive. Thus, suppose that there is some class A with $\forall a \in A.\ \exists a' \in A.\ a' R^+ a$. We have to show that A is empty. Set $B := \{\ b \mid \exists a \in A.\ aR^* b\ \}$.
Claim 1: For any $b \in B$, there is some $b' \in B$ with $b' R b$.
Proof of Claim 1: By definition of B and the property of A, there is some $a \in A$ with $aR^+ b$. Thus, there is some b' with $aR^* b' R b$. Q.e.d. (Claim 1)
By Claim 1 and the assumption that R is wellfounded, we get $A \subseteq B = \emptyset$.
Q.e.d. (Lemma 3.1)

Proof of Lemma 4.5

4.2 ⇒ 4.4: Let $<$ be an ordering with $\forall a \in A.\ \exists a' \in A.\ a > a'$. Set $R := > \cap\, (A \times A)$. Now $\mathrm{dom}(R) = A \supseteq \mathrm{ran}(R)$. Assume $A \neq \emptyset$. By the Principle of Dependent Choice, R is not terminating. This contradicts Version 1 and Version 2 of the Principle of Descente Infinie.

4.4 ⇒ 4.2: Let R be a binary relation with $\mathrm{ran}(R) \subseteq \mathrm{dom}(R) \neq \emptyset$. We are going to show that R is not terminating.
Set $A := \left\{\ a \ \middle|\ \exists n \in \mathbf{N}.\ \left(\begin{array}{c} a : \{0,\ldots,n\} \to \mathrm{dom}(R) \\ \wedge\ \ \forall i \prec n.\ a_i R a_{i+1} \end{array}\right) \right\}$.
Define $\lesssim$ on A by $a \gtrsim a'$ if $\left(\begin{array}{c} \mathrm{dom}(a) \subseteq \mathrm{dom}(a') \\ \wedge\ \ \forall i \in \mathrm{dom}(a).\ a_i = a'_i \end{array}\right)$. Let $<$ be the ordering of $\lesssim$.
Claim 1: $\forall a \in A.\ \exists a' \in A.\ a > a'$.
Proof of Claim 1: For $a : \{0,\ldots,n\} \to \mathrm{dom}(R)$ we have to show the existence of some $a' : \{0,\ldots,n,n{+}1\} \to \mathrm{dom}(R)$ with $a > a'$. When we set $a'_i := a_i$ for $i \preceq n$ then (due to $a_n \in \mathrm{dom}(R)$) there exists an a'_{n+1} with $a_n\ R\ a'_{n+1}$, and then $a'_{n+1} \in \mathrm{ran}(R) \subseteq \mathrm{dom}(R)$.
Q.e.d. (Claim 1)
Since $\mathrm{dom}(R) \neq \emptyset$ we have $A \neq \emptyset$. Thus, by Claim 1 and the Principle of Descente Infinie (Version 1) there is some non-terminating sequence $(a_i)_{i \in \mathbf{N}}$ in $>$ and we set $C := \mathrm{ran}(a)$ or (Version 2) there is some $C \subseteq A$ totally ordered by $<$ that has no $<$-minimal element. But then $\bigcup C$ is a non-terminating sequence in R.

4.4(Version 1) ⇒ 4.3: If $<$ is not wellfounded, then there is some non-empty class A with $\forall a \in A.\ \exists a' \in A.\ a > a'$. Thus, by the Principle of Descente Infinie, $> \cap\, (A \times A)$ is not terminating, which implies that $>$ is not terminating.

4.3 ⇒ 4.4(Version 1): Let $<$ be an ordering. Then $< \cap\, (A \times A)$ is an ordering, too. Thus, if $> \cap\, (A \times A)$ is terminating, by the Principle of Wellfoundedness, $< \cap\, (A \times A)$ is a wellfounded ordering. In case of $\forall a \in A.\ \exists a' \in A.\ a > a'$, this means that A must be empty. **Q.e.d. (Lemma 4.5)**

Proof of Lemma 9.3

Set $\lhd := (R \cup S_e)^+$ and $S_\pi := \lhd \cap (\mathrm{V}_{\forall,\mathrm{w}} \times \mathrm{V}_{\forall,\mathrm{s}})$. As e is an existential $(\mathcal{A}, R)$-valuation, $\lhd$ is a wellfounded ordering. With the help of a choice function and by induction on $y^{\forall,\mathrm{s}} \in \mathrm{V}_{\forall,\mathrm{s}}$ in $\lhd$ we can define $\pi(y^{\forall,\mathrm{s}}) \in (S_\pi\langle\!\{y^{\forall,\mathrm{s}}\}\!\rangle \to \mathcal{A}) \to \mathcal{A}$ in the following way:
Let $\tau \in S_\pi\langle\!\{y^{\forall,\mathrm{s}}\}\!\rangle \to \mathcal{A}$. In case of $y^{\forall,\mathrm{s}} \in \mathrm{V}_{\forall,\mathrm{s}} \backslash \mathrm{dom}(C)$ we choose an arbitrary value for $\pi(y^{\forall,\mathrm{s}})(\tau)$ from the universe of $\mathcal{A}$ (of the appropriate type). Note that universes are assumed to be non-empty, cf. Section 3.2.
In case of $y^{\forall,\mathrm{s}} \in \mathrm{dom}(C)$, we have the following situation: $C(y^{\forall,\mathrm{s}}) = \lambda v_0.\ \ldots \lambda v_{l-1}.\ B$, B is a formula whose unbound variables from $\mathrm{V}_{\mathrm{bound}}$ are among $\{v_0, \ldots, v_{l-1}\} \subseteq \mathrm{V}_{\mathrm{bound}}$ and where, for $v_0 : \alpha_0, \ldots, v_{l-1} : \alpha_{l-1}$, we have $y^{\forall,\mathrm{s}} : \alpha_0 \to \ldots \to \alpha_{l-1} \to \alpha_l$ for some type α_l, and any occurrence of $y^{\forall,\mathrm{s}}$ in B is of the form $y^{\forall,\mathrm{s}}(v_0) \cdots (v_{l-1})$. In this case, we let $\pi(y^{\forall,\mathrm{s}})(\tau)$ be the function f that for $\chi \in \{v_0, \ldots, v_{l-1}\} \to \mathcal{A}$ chooses a value from universe of $\mathcal{A}$ for $f(\chi(v_0)) \cdots (\chi(v_{l-1}))$ such that, if possible, B is $(\epsilon(\pi)(\tau \uplus \tau') \uplus \tau \uplus \tau' \uplus \chi, e, \mathcal{A})$-valid for an arbitrary $\tau' \in (\mathrm{V}_{\forall,\mathrm{w}} \backslash \mathrm{dom}(\tau)) \to \mathcal{A}$. Note that this definition of $f(\chi(v_0)) \cdots (\chi(v_{l-1}))$ does not depend on the values of $f(\chi'(v_0)) \cdots (\chi'(v_{l-1}))$ for a different $\chi' \in \{v_0, \ldots, v_{l-1}\} \to \mathcal{A}$ because any occurrence of $y^{\forall,\mathrm{s}}$ in B is of the form $y^{\forall,\mathrm{s}}(v_0) \cdots (v_{l-1})$.

<u>Claim 1:</u> For $z^\forall \in \mathrm{V}_\forall$ with $z^\forall \lhd y^{\forall,\mathrm{s}}$, $(\epsilon(\pi)(\tau \uplus \tau') \uplus \tau \uplus \tau')(z^\forall)$ depends only τ, $\pi(z^\forall)$, and $z^\forall$.
<u>Claim 2:</u> For $x^\exists \in \mathrm{V}_\exists$ with $z^\exists \lhd y^{\forall,\mathrm{s}}$, $\epsilon(e)(\epsilon(\pi)(\tau \uplus \tau') \uplus \tau \uplus \tau')(x^\exists)$ depends only τ, ${}_{\lhd\langle\!\{y^{\forall,\mathrm{s}}\}\!\rangle}{\upharpoonleft}\pi$ and $e(x^\exists)$.
<u>Claim 3:</u> The definition of $\pi(y^{\forall,\mathrm{s}})(\tau)$ depends only on such $\pi(v^{\forall,\mathrm{s}})$ with $v^{\forall,\mathrm{s}} \lhd y^{\forall,\mathrm{s}}$.
<u>Claim 4:</u> The definition of $\pi(y^{\forall,\mathrm{s}})(\tau)$ does not depend on τ'.

<u>Proof of Claim 1:</u> For $z^\forall \in \mathrm{V}_{\forall,\mathrm{w}}$ we have $(\epsilon(\pi)(\tau \uplus \tau') \uplus \tau \uplus \tau')(z^\forall) = \tau(z^\forall)$ due to $z^\forall \in S_\pi\langle\!\{y^{\forall,\mathrm{s}}\}\!\rangle$. Moreover, for $z^\forall \in \mathrm{V}_{\forall,\mathrm{s}}$, we have $S_\pi\langle\!\{z^\forall\}\!\rangle \subseteq S_\pi\langle\!\{y^{\forall,\mathrm{s}}\}\!\rangle$, and therefore $(\epsilon(\pi)(\tau \uplus \tau') \uplus \tau \uplus \tau')(z^\forall) = \pi(z^\forall)({}_{S_\pi\langle\!\{z^\forall\}\!\rangle}{\upharpoonleft}(\tau \uplus \tau')) = \pi(z^\forall)({}_{S_\pi\langle\!\{z^\forall\}\!\rangle}{\upharpoonleft}\tau)$. <u>Q.e.d. (Claim 1)</u>

<u>Proof of Claim 2:</u> As $S_e\langle\!\{x^\exists\}\!\rangle \subseteq \lhd\langle\!\{y^{\forall,\mathrm{s}}\}\!\rangle$, and $\epsilon(e)(\epsilon(\pi)(\tau \uplus \tau') \uplus \tau \uplus \tau')(x^\exists) = e(x^\exists)({}_{S_e\langle\!\{x^\exists\}\!\rangle}{\upharpoonleft}(\epsilon(\pi)(\tau \uplus \tau') \uplus \tau \uplus \tau'))$ this follows from Claim 1. <u>Q.e.d. (Claim 2)</u>

<u>Proof of Claim 3 and 4:</u> Since C is an R-choice-condition, we have $z \lhd y^{\forall,\mathrm{s}}$ for all $z \in \mathcal{V}_{\mathrm{free}}(C(y^{\forall,\mathrm{s}})) \backslash \{y^{\forall,\mathrm{s}}\}$. Thus, this follows from Claim 1 and Claim 2. <u>Q.e.d. (Claim 3, 4)</u>

Now π is well-defined by Claim 3 and Claim 4 and obviously semantical. Thus, item 1 of Definition 9.2 is satisfied because $(R \cup S_e \cup S_\pi)^+ = \lhd$ is a wellfounded ordering. For showing item 2, let $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$, $y^{\forall,\mathrm{s}} \in \mathrm{dom}(C)$, and $C(y^{\forall,\mathrm{s}}) = \lambda v_0.\ \ldots \lambda v_{l-1}.\ B$, and assume to the contrary that, for some $\eta \in \{y^{\forall,\mathrm{s}}\} \to \mathcal{A}$ and $\chi \in \{v_0, \ldots, v_{l-1}\} \to \mathcal{A}$, B is $({}_{\mathrm{V}_{\forall,\mathrm{s}} \backslash \{y^{\forall,\mathrm{s}}\}}{\upharpoonleft}(\epsilon(\pi)(\tau)) \uplus \eta \uplus \tau \uplus \chi, e, \mathcal{A})$-valid, but not $(\epsilon(\pi)(\tau) \uplus \tau \uplus \chi, e, \mathcal{A})$-valid. This contradicts the definition of $\pi(y^{\forall,\mathrm{s}})({}_{S_\pi\langle\!\{y^{\forall,\mathrm{s}}\}\!\rangle}{\upharpoonleft}\tau)$ from above due to Claim 4. **Q.e.d. (Lemma 9.3)**

Proof of Lemma 9.6

Here we denote concatenation (product) of relations '$\circ$' simply by juxtaposition and assume it to have higher priority than any other binary operator. R'^{+} is a wellfounded ordering simply because R' is the σ-update of R and σ is an R-substitution. Now it suffices to show the following two claims for an arbitrary $y^{\forall,\mathrm{s}} \in \mathrm{dom}(C')$:

Claim 1: For all $z^{\forall} \in \mathcal{V}_{\forall}(C'(y^{\forall,\mathrm{s}}))\backslash\{y^{\forall,\mathrm{s}}\}$: $z^{\forall} R'^{+} y^{\forall,\mathrm{s}}$.

Claim 2: For all $u^{\exists} \in \mathcal{V}_{\exists}(C'(y^{\forall,\mathrm{s}}))$: $u^{\exists} R'^{+} y^{\forall,\mathrm{s}}$.

Proof of Claim 1: Let $z^{\forall} \in \mathcal{V}_{\forall}(C'(y^{\forall,\mathrm{s}}))\backslash\{y^{\forall,\mathrm{s}}\}$. By the definition of C' this means $z^{\forall} \in \mathcal{V}_{\forall}(C(y^{\forall,\mathrm{s}}))\backslash\{y^{\forall,\mathrm{s}}\}$ or there is some $u \in \mathcal{V}(C(y^{\forall,\mathrm{s}}))$ with $z^{\forall}\ U_{\sigma}\ u$. Since C is an R-choice-condition, we have $z^{\forall}\ R^{+}\ y^{\forall,\mathrm{s}}$ or $z^{\forall}\ U_{\sigma}\ u\ R^{*}\ y^{\forall,\mathrm{s}}$. As R' is the σ-update of R, we have $R \cup U_{\sigma} \subseteq R'$.[28] Thus $z^{\forall} R'^{+} y^{\forall,\mathrm{s}}$. Q.e.d. (Claim 1)

Proof of Claim 2: Let $u^{\exists} \in \mathcal{V}_{\exists}(C'(y^{\forall,\mathrm{s}}))$. By the definition of C' there is some $v \in \mathcal{V}(C(y^{\forall,\mathrm{s}}))$ with $u^{\exists}\ ({}_{\mathrm{V}_{\exists}\backslash\mathrm{dom}(\sigma)}\!\upharpoonleft\!\mathrm{id} \cup E_{\sigma})\ v$. Since C is an R-choice-condition, we have $v\ R^{*}\ y^{\forall,\mathrm{s}}$, i.e. $u^{\exists}\ ({}_{\mathrm{V}_{\exists}\backslash\mathrm{dom}(\sigma)}\!\upharpoonleft\!\mathrm{id} \cup E_{\sigma}) R^{*}\ y^{\forall,\mathrm{s}}$. As R' is the σ-update of R, we have $({}_{\mathrm{V}_{\exists}\backslash\mathrm{dom}(\sigma)}\!\upharpoonleft\!\mathrm{id} \cup E_{\sigma}) R^{*} \subseteq (R \cup E_{\sigma})^{*} \subseteq R'^{*}$.[29] Thus $u^{\exists} R'^{+} y^{\forall,\mathrm{s}}$. Q.e.d. (Claim 2) **Q.e.d. (Lemma 9.6)**

Proof of Lemma 10.2

As G is $\mathrm{V}_{\exists}\times\mathrm{V}_{\forall}$-valid in $\mathcal{A}$, there is some existential $(\mathcal{A}, \mathrm{V}_{\exists}\times\mathrm{V}_{\forall})$-valuation e s.t. G is $(e, \mathcal{A})$-valid.

Claim 1: e is an existential $(\mathcal{A}, R)$-valuation.

Proof of Claim 1: As e is an existential $(\mathcal{A}, \mathrm{V}_{\exists}\times\mathrm{V}_{\forall})$-valuation, $S_e\circ(\mathrm{V}_{\exists}\times\mathrm{V}_{\forall})$ is irreflexive. This means $S_e = \emptyset$, i.e. $(R \cup S_e)^{+} = R^{+}$. As C is an R-choice-condition, R^{+} is a wellfounded ordering. This means that $(R \cup S_e)^{+}$ is a wellfounded ordering, as was to be shown. Q.e.d. (Claim 1)

By Claim 1, G is immediately R-valid. Moreover, by Claim 1 and Lemma 9.3, there is some π that is $(e, \mathcal{A})$-compatible with (C, R). As G is $(e, \mathcal{A})$-valid, G is also $(\epsilon(\pi)(\tau) \uplus \tau, e, \mathcal{A})$-valid for all $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$. Then, as π is $(e, \mathcal{A})$-compatible with (C, R), G is (C, R)-strongly $(e, \mathcal{A})$-valid. Then, by Claim 1, G is (C, R)-strongly valid in $\mathcal{A}$. **Q.e.d. (Lemma 10.2)**

Proof of Lemma 10.3

As G is (C, R)-strongly valid in $\mathcal{A}$, there are some existential $(\mathcal{A}, R)$-valuation e and some π s.t. π is $(e, \mathcal{A})$-compatible with (C, R) and G is strongly $(\pi, e, \mathcal{A})$-valid.

Set $S_{e'} := ({}_{\mathrm{V}_{\forall,\mathrm{w}}}\!\upharpoonleft\!\mathrm{id} \cup S_{\pi}) \circ S_e\!\upharpoonright_{\mathrm{V}_{\exists}\backslash\mathrm{ran}(\varsigma)} \ \cup\ S_{\pi} \circ \varsigma$. We define e' via:

For $x^{\exists} \in \mathrm{V}_{\exists}\backslash\mathrm{ran}(\varsigma)$: For $\tau \in S_{e'}\langle\{x^{\exists}\}\rangle \to \mathcal{A}$:

$$e'(x^{\exists})(\tau) := e(x^{\exists})({}_{S_e\langle\{x^{\exists}\}\rangle}\!\upharpoonleft\!(\epsilon(\pi)(\tau \uplus \tau') \uplus \tau \uplus \tau'))$$

where $\tau' \in (\mathrm{V}_{\forall,\mathrm{w}}\backslash\mathrm{dom}(\tau)) \to \mathcal{A}$. Note that this right-hand side is okay because $\mathrm{dom}(\tau) \subseteq \mathrm{V}_{\forall,\mathrm{w}}$; indeed, due to $x^{\exists} \notin \mathrm{ran}(\varsigma)$, we have $S_{e'}\langle\{x^{\exists}\}\rangle = (\mathrm{V}_{\forall,\mathrm{w}} \cap S_e\langle\{x^{\exists}\}\rangle) \cup (S_{\pi} \circ S_e)\langle\{x^{\exists}\}\rangle \subseteq \mathrm{V}_{\forall,\mathrm{w}}$. Furthermore, note that this right-hand side does not depend on τ' because $\mathrm{V}_{\forall,\mathrm{w}} \cap S_e\langle\{x^{\exists}\}\rangle \subseteq S_{e'}\langle\{x^{\exists}\}\rangle = \mathrm{dom}(\tau)$, and for $y^{\forall,\mathrm{s}} \in S_e\langle\{x^{\exists}\}\rangle$, we have $S_{\pi}\langle\{y^{\forall,\mathrm{s}}\}\rangle \subseteq (S_{\pi} \circ S_e)\langle\{x^{\exists}\}\rangle \subseteq S_{e'}\langle\{x^{\exists}\}\rangle$ and therefore $\epsilon(\pi)(\tau \uplus \tau')(y^{\forall,\mathrm{s}}) = \pi(y^{\forall,\mathrm{s}})({}_{S_{\pi}\langle\{y^{\forall,\mathrm{s}}\}\rangle}\!\upharpoonleft\!(\tau \uplus \tau')) = \pi(y^{\forall,\mathrm{s}})({}_{S_{\pi}\langle\{y^{\forall,\mathrm{s}}\}\rangle}\!\upharpoonleft\!\tau)$.

For $x^{\exists} \in \mathrm{ran}(\varsigma)$: For $\tau \in S_{e'}\langle\{x^{\exists}\}\rangle \to \mathcal{A}$:

$$e'(x^{\exists})(\tau) := \pi(\varsigma^{-1}(x^{\exists}))(\tau).$$

Note that this right-hand side is okay because, due to $x^{\exists} \in \mathrm{ran}(\varsigma)$, we have $S_{e'}\langle\{x^{\exists}\}\rangle = S_{\pi}\langle\{\varsigma^{-1}(x^{\exists})\}\rangle \subseteq \mathrm{V}_{\forall,\mathrm{w}}$.

Claim 1: e' is an existential $(\mathcal{A}, R')$-valuation.
Proof of Claim 1: Here we denote concatenation (product) of relations '$\circ$' simply by juxtaposition and assume it to have higher priority than any other binary operator. As π is $(e, \mathcal{A})$-compatible with (C, R), we know that $(R \cup S_e \cup S_\pi)^+$ is a wellfounded ordering. Thus, its subset
$\left(\ ({}_{\mathrm{V}_{\forall,\mathrm{w}} \cup \mathrm{V}_\exists \setminus \mathrm{ran}(\varsigma)}\rceil \mathrm{id} \cup S_\pi \varsigma \varsigma^{-1}) R^+ \upharpoonright_{\mathrm{V}_{\forall,\mathrm{w}} \cup \mathrm{V}_\exists \setminus \mathrm{ran}(\varsigma)} \ \cup\ ({}_{\mathrm{V}_{\forall,\mathrm{w}}}\rceil \mathrm{id} \cup S_\pi) S_e \upharpoonright_{\mathrm{V}_\exists \setminus \mathrm{ran}(\varsigma)}\ \right)^+$
is a wellfounded ordering, too.
Since the domain of this relation and $\mathrm{dom}(S_\pi)$ are disjoint from $\mathrm{ran}(\varsigma)$, we know that
$\left(\ ({}_{\mathrm{V}_{\forall,\mathrm{w}} \cup \mathrm{V}_\exists \setminus \mathrm{ran}(\varsigma)}\rceil \mathrm{id} \cup S_\pi \varsigma \varsigma^{-1}) R^+ \upharpoonright_{\mathrm{V}_{\forall,\mathrm{w}} \cup \mathrm{V}_\exists \setminus \mathrm{ran}(\varsigma)} \ \cup\ ({}_{\mathrm{V}_{\forall,\mathrm{w}}}\rceil \mathrm{id} \cup S_\pi) S_e \upharpoonright_{\mathrm{V}_\exists \setminus \mathrm{ran}(\varsigma)} \ \cup\ S_\pi \varsigma\ \right)^+$
is a wellfounded ordering, too.
Since the domain of this relation and $\mathrm{V}_\exists$ are disjoint from $\mathrm{V}_{\forall,\mathrm{s}}$, we know that
$\left(\ ({}_{\mathrm{V}_{\forall,\mathrm{w}} \cup \mathrm{V}_\exists \setminus \mathrm{ran}(\varsigma)}\rceil \mathrm{id} \cup S_\pi \varsigma \varsigma^{-1}) R^+ \upharpoonright_{\mathrm{V}_{\forall,\mathrm{w}} \cup \mathrm{V}_\exists \setminus \mathrm{ran}(\varsigma)} \cup \mathrm{V}_\exists \times \mathrm{V}_{\forall,\mathrm{s}} \cup ({}_{\mathrm{V}_{\forall,\mathrm{w}}}\rceil \mathrm{id} \cup S_\pi) S_e \upharpoonright_{\mathrm{V}_\exists \setminus \mathrm{ran}(\varsigma)} \cup S_\pi \varsigma\ \right)^+$
is a wellfounded ordering, too. Since a step inside the transitive closure of the previous term that can precede ς^{-1} can only be a step with $S_\pi \varsigma$ (due to $\mathrm{dom}(\varsigma^{-1}) = \mathrm{ran}(\varsigma) \subseteq \mathrm{V}_\exists$),
$\left(\ ({}_{\mathrm{V}_{\forall,\mathrm{w}} \cup \mathrm{V}_\exists \setminus \mathrm{ran}(\varsigma)}\rceil \mathrm{id} \cup \varsigma^{-1}) R^+ \upharpoonright_{\mathrm{V}_{\forall,\mathrm{w}} \cup \mathrm{V}_\exists \setminus \mathrm{ran}(\varsigma)} \ \cup\ \mathrm{V}_\exists \times \mathrm{V}_{\forall,\mathrm{s}} \ \cup\ ({}_{\mathrm{V}_{\forall,\mathrm{w}}}\rceil \mathrm{id} \cup S_\pi) S_e \upharpoonright_{\mathrm{V}_\exists \setminus \mathrm{ran}(\varsigma)} \ \cup\ S_\pi \varsigma\ \right)^+$
is a wellfounded ordering, too. The latter relation is a superset of $(R' \cup S_{e'})^+$ which we had to show to be a wellfounded ordering. Q.e.d. (Claim 1)

As the universes are assumed to be non-empty (cf. Section 3.2), there is some $\delta \in \mathrm{V}_{\forall,\mathrm{s}} \to \mathcal{A}$ by the Axiom of Choice. Define π' by $\pi'(y^{\forall,\mathrm{s}})(\emptyset) := \delta(y^{\forall,\mathrm{s}})$.

Claim 2: π' is $(e', \mathcal{A})$-compatible with $(\emptyset, R')$.
Proof of Claim 2: We have $S_{\pi'} = \emptyset$. Thus, $(R' \cup S_{e'} \cup S_{\pi'})^+$ is equal to $(R' \cup S_{e'})^+$, which is a wellfounded ordering by Claim 1. Q.e.d. (Claim 2)

Claim 3: For $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$ and $x^\exists \in \mathcal{V}_\exists(G)$: $\epsilon(e')(\epsilon(\pi')(\tau) \uplus \tau)(x^\exists) = \epsilon(e)(\epsilon(\pi)(\tau) \uplus \tau)(x^\exists)$.
Proof of Claim 3: We have $x^\exists \in \mathcal{V}_\exists(G) \subseteq \mathrm{V}_\exists \setminus \mathrm{ran}(\varsigma)$. Thus, by the discussion of the first case of the definition of e', we have $\epsilon(e')(\epsilon(\pi')(\tau) \uplus \tau)(x^\exists) = e'(x^\exists)({}_{S_{e'}\langle\!\{x^\exists\}\!\rangle}\rceil \tau) = e(x^\exists)({}_{S_e\langle\!\{x^\exists\}\!\rangle}\rceil(\epsilon(\pi)(\tau) \uplus \tau)) = \epsilon(e)(\epsilon(\pi)(\tau) \uplus \tau)(x^\exists)$. Q.e.d. (Claim 3)

Claim 4: For $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$ and $y^{\forall,\mathrm{s}} \in \mathcal{V}_{\forall,\mathrm{s}}(G)$: $\epsilon(e')(\epsilon(\pi')(\tau) \uplus \tau)(\varsigma(y^{\forall,\mathrm{s}})) = \epsilon(\pi)(\tau)(y^{\forall,\mathrm{s}})$.
Proof of Claim 4: Since $\varsigma(y^{\forall,\mathrm{s}}) \in \mathrm{ran}(\varsigma)$, by the discussion of the second case of the definition of e', we have $\epsilon(e')(\epsilon(\pi')(\tau) \uplus \tau)(\varsigma(y^{\forall,\mathrm{s}})) = e'(\varsigma(y^{\forall,\mathrm{s}}))({}_{S_{e'}\langle\!\{\varsigma(y^{\forall,\mathrm{s}})\}\!\rangle}\rceil \tau) = \pi(\varsigma^{-1}(\varsigma(y^{\forall,\mathrm{s}})))({}_{S_\pi\langle\!\{\varsigma^{-1}(\varsigma(y^{\forall,\mathrm{s}}))\}\!\rangle}\rceil \tau) = \pi(y^{\forall,\mathrm{s}})({}_{S_\pi\langle\!\{y^{\forall,\mathrm{s}}\}\!\rangle}\rceil \tau) = \epsilon(\pi)(\tau)(y^{\forall,\mathrm{s}})$. Q.e.d. (Claim 4)

Claim 5: $G\varsigma$ is strongly $(\pi', e', \mathcal{A})$-valid.
Proof of Claim 5: Let $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$ be arbitrary. First by the Substitution-Lemma, second by Claim 3, $\mathcal{V}_{\forall,\mathrm{s}}(G) \subseteq \mathrm{dom}(\varsigma)$, Claim 4, and third as G is strongly $(\pi, e, \mathcal{A})$-valid, we get:
$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\epsilon(\pi')(\tau) \uplus \tau) \uplus \epsilon(\pi')(\tau) \uplus \tau)(G\varsigma) =$

$$\mathrm{eval}\left(\begin{array}{l} \quad \mathcal{A} \ \uplus\ \epsilon(e')(\epsilon(\pi')(\tau) \uplus \tau) \\ \uplus\ {}_{\mathrm{V}_{\forall,\mathrm{s}} \setminus \mathrm{dom}(\varsigma)}\rceil(\epsilon(\pi')(\tau)) \ \uplus\ \varsigma \circ (\epsilon(e')(\epsilon(\pi')(\tau) \uplus \tau)) \ \uplus\ \tau \end{array}\right)(G) =$$

$\mathrm{eval}(\ \mathcal{A} \uplus \epsilon(e)(\epsilon(\pi)(\tau) \uplus \tau) \uplus \epsilon(\pi)(\tau) \uplus \tau\ )(G) = \mathsf{TRUE}$ Q.e.d. (Claim 5)

Claim 6: $G\varsigma$ is R'-valid in $\mathcal{A}$.
Proof of Claim 6: First note that by Claim 1, e' is an existential $(\mathcal{A}, R')$-valuation. Let $\tau' \in \mathrm{V}_\forall \to \mathcal{A}$ be arbitrary. When we set $\delta := {}_{\mathrm{V}_{\forall,\mathrm{s}}}\rceil \tau'$ and $\tau := {}_{\mathrm{V}_{\forall,\mathrm{w}}}\rceil \tau'$, we get
$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\tau') \uplus \tau')(G\varsigma) = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\epsilon(\pi')(\tau) \uplus \tau) \uplus \epsilon(\pi')(\tau) \uplus \tau)(G\varsigma) = \mathsf{TRUE}$, where the latter step is due to Claim 5. Q.e.d. (Claim 6)

Now we conclude that $G\varsigma$ is $(\emptyset, R')$-strongly valid in $\mathcal{A}$ (by Claim 1, Claim 2, and Claim 5) and R'-valid (Claim 6) in $\mathcal{A}$. **Q.e.d. (Lemma 10.3)**

Proof of Lemma 11.2

(1), (2), (3), and (4) are trivial.

(5): Let e be an existential $(\mathcal{A}, R')$-valuation and π be $(e, \mathcal{A})$-compatible with (C', R'). Due to $R \subseteq R'$, by Lemma A.1(1), e is an existential $(\mathcal{A}, R)$-valuation, too.

Claim 0: π is $(e, \mathcal{A})$-compatible with (C, R).
Proof of Claim 0: As π is $(e, \mathcal{A})$-compatible with (C', R'), and $C \subseteq C'$, π is $(e, \mathcal{A})$-compatible with (C, R'). As $R \subseteq R'$, π is $(e, \mathcal{A})$-compatible with (C, R). Q.e.d. (Claim 0)

(5a): As G_0 is (C', R')-strongly valid in $\mathcal{A}$, there is an existential $(\mathcal{A}, R')$-valuation e and some π s.t. π is $(e, \mathcal{A})$-compatible with (C', R') and G_0 is strongly $(\pi, e, \mathcal{A})$-valid. Thus, by Claim 0, G_0 is (C, R)-strongly $(e, \mathcal{A})$-valid. As e is an existential $(\mathcal{A}, R)$-valuation, G_0 is (C, R)-strongly valid in $\mathcal{A}$.

(5b): Let e and π be given as in (5) above and suppose that G_1 is strongly $(\pi, e, \mathcal{A})$-valid. Thus, by Claim 0 and since e is an existential $(\mathcal{A}, R)$-valuation and G_0 (C, R)-reduces to G_1, also G_0 is strongly $(\pi, e, \mathcal{A})$-valid as was to be shown.

(6a): As $G_0\sigma \cup \langle O \rangle Q_{C,\sigma}$ is (C', R')-strongly valid in $\mathcal{A}$, there is an existential $(\mathcal{A}, R')$-valuation e' and some π' s.t. π' is $(e', \mathcal{A})$-compatible with (C', R') and $G_0\sigma \cup \langle O \rangle Q_{C,\sigma}$ is strongly $(\pi', e', \mathcal{A})$-valid. Let e and π be given as in Lemma A.2. Then G_0 is strongly $(\pi, e, \mathcal{A})$-valid. Moreover, as π is $(e, \mathcal{A})$-compatible with (R, C) and as e is an existential $(\mathcal{A}, R)$-valuation, G_0 is (C, R)-strongly valid in $\mathcal{A}$.

(6b): Let e' be an existential $(\mathcal{A}, R')$-valuation, π' be $(e', \mathcal{A})$-compatible with (C', R'), and suppose that $G_1\sigma \cup \langle O \rangle Q_{C,\sigma}$ is strongly $(\pi', e', \mathcal{A})$-valid. Let π and the existential $(\mathcal{A}, R)$-valuation e be given as in Lemma A.2. Then π is $(e, \mathcal{A})$-compatible with (C, R), and G_1 is strongly $(\pi, e, \mathcal{A})$-valid. By assumption, G_0 strongly (C, R)-reduces to G_1. Thus, G_0 is strongly $(\pi, e, \mathcal{A})$-valid, too. By Lemma A.2(2), this means that $G_0\sigma$ is strongly $(\pi', e', \mathcal{A})$-valid as was to be shown. **Q.e.d. (Lemma 11.2)**

Proof of Lemma 12.3

(1), (2), (3), and (4) are trivial.

(5): Let $\mathcal{A} \in \mathrm{K}$, $S \in G_0$, let e be an existential $(\mathcal{A}, R')$-valuation, and π be $(e, \mathcal{A})$-compatible with (C', R'). Suppose that (S, τ) is an $(\pi, e, \mathcal{A})$-counterexample. Due to $R \subseteq R'$, by Lemma A.1(1), e is an existential $(\mathcal{A}, R)$-valuation, too. As π is $(e, \mathcal{A})$-compatible with (C', R'), and $C \subseteq C'$, π is $(e, \mathcal{A})$-compatible with (C, R'). As $R \subseteq R'$, π is $(e, \mathcal{A})$-compatible with (C, R). By assumption, $G_0 \curvearrowright_{C,R} (G_1, L_1)$. Thus, there is some $(\pi, e, \mathcal{A})$-counterexample (S', τ') with $S' \in L_1$ or $S' \in G_1$ and in the latter case the long conjunction of the definition of foundedness as was to be shown.

(6): Let $\mathcal{A} \in \mathrm{K}$, let e' be an existential $(\mathcal{A}, R')$-valuation, and π' be $(e', \mathcal{A})$-compatible with (C', R'). Let $(\Gamma, (w, <, \lesssim)) \in G_0$ and assume that $((\Gamma\sigma, (w\sigma, <\sigma, \lesssim\sigma)), \tau)$ is an $(\pi', e', \mathcal{A})$-counterexample. Assuming that there is no $(\pi', e', \mathcal{A})$-counterexample of $L_1\sigma \cup L_2$, we have to find some $(\pi', e', \mathcal{A})$-counterexample $((\Gamma'\sigma, (w'\sigma, <'\sigma, \lesssim'\sigma)), \tau')$ with $(\Gamma', (w', <', \lesssim')) \in G_1$, plus the long conjunction of the definition of foundedness for e, π replaced with e', π', resp., and $<$, $\lesssim$, w, $<'$, $\lesssim'$, w' replaced with their σ-instantiations. By our assumption on no $(\pi', e', \mathcal{A})$-counterexamples of L_2, we can apply Lemma A.2 to get a an existential $(\mathcal{A}, R)$-valuation e and a π that is $(e, \mathcal{A})$-compatible with (C, R). Moreover, by this lemma, $((\mathit{\Gamma}, (w, <, \lesssim)), \tau)$ is an $(\pi, e, \mathcal{A})$-counterexample. By assumption, $G_0 \curvearrowright_{C,R} (G_1, L_1)$. Thus, there is some $(\pi, e, \mathcal{A})$-counterexample $((\mathit{\Gamma}', (w', <', \lesssim')), \tau')$ with $(\mathit{\Gamma}', (w', <', \lesssim')) \in L_1$ or $(\mathit{\Gamma}', (w', <', \lesssim')) \in G_1$, plus in the latter case the long conjunction of the definition of foundedness. By Lemma A.2, this means that $((\mathit{\Gamma}'\sigma, (w'\sigma, <'\sigma, \lesssim'\sigma)), \tau')$ is an $(\pi', e', \mathcal{A})$-counterexample. We only have left to show that in the latter case the long conjunction of the definition of foundedness holds when we replace e, π with e', π', resp., and $<$, $\lesssim$, w, $<'$, $\lesssim'$, w' with their σ-instantiations. This is nearly implied by Lemma A.2(1); the only problem is that $<$, $\lesssim$, $<'$, $\lesssim'$ are possibly no terms (so that the B of Lemma A.2(1) cannot be instantiated with them). Thus, for arbitrary $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \rightarrow \mathcal{A}$ and δ and δ' given as in Lemma A.2(1), we still have to prove say $\mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \delta')(<\sigma) = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(<)$. After expanding the shorthand on both sides for some distinct $x, y \in \mathrm{V}_{\mathrm{bound}} \setminus \mathcal{V}(<, \mathrm{dom}(\sigma), \mathrm{ran}(\sigma))$, this follows from

$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \delta' \uplus \{x{\mapsto}a,\ y{\mapsto}b\})(x\ (<\sigma)\ y) =$ (as $x, y \notin \mathrm{dom}(\sigma)$)
$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \delta' \uplus \{x{\mapsto}a,\ y{\mapsto}b\})((x < y)\sigma) =$ (due to Lemma A.2.1)
$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta \uplus \{x{\mapsto}a,\ y{\mapsto}b\})(x < y)$.

(7): Let $\mathcal{A} \in \mathrm{K}$, e be some existential $(\mathcal{A}, R)$-valuation, and π be $(e, \mathcal{A})$-compatible with (C, R). Let D be the class of $(\pi, e, \mathcal{A})$-counterexamples (S, τ) with $S \in H_1$ for that there is no $(\pi, e, \mathcal{A})$-counterexample (S', τ') with $S' \in L_1$ or $S' \in G_1$, and in the latter case the long existential quantification of the definition of foundedness with the second alternative being valid. It suffices to show that D is empty because this means $H_1 \curvearrowright_{C,R} (G_1, L_1)$. To the contrary, suppose $((\mathit{\Gamma}''', (w''', <''', \lesssim''')), \tau''') \in D$ and set $\delta''' := \epsilon(\pi)(\tau''') \uplus \tau'''$, $\lhd := \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta''') \uplus \delta''')(<''')$, and $\mathrel{\underset{\sim}{\lhd}} := \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta''') \uplus \delta''')(\lesssim''')$. Set

$$A \ := \left\{ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(w) \;\middle|\; \left(\begin{array}{ll} & ((\mathit{\Gamma}, (w, <, \lesssim)), \tau) \in D \\ \wedge & \delta = \epsilon(\pi)(\tau) \uplus \tau \\ \wedge & \lhd = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(<) \\ \wedge & \mathrel{\underset{\sim}{\lhd}} = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(\lesssim) \end{array} \right) \right\}.$$

Then $\mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta''') \uplus \delta''')(w''') \in A$. Due to the assumed $H_1 \searrow\!/\!\curvearrowright_{C,R} (H_1, G_1, L_1)$ and $D \neq \emptyset$, $\lhd^+$ is a wellfounded ordering. Thus, A must have a $\lhd^+$-minimal element $\bar{w}$. Thus, there must be some $((\mathit{\Gamma}, (w, <, \lesssim)), \tau) \in D$ with $\delta = \epsilon(\pi)(\tau) \uplus \tau$, $\lhd = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(<)$, $\mathrel{\underset{\sim}{\lhd}} = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(\lesssim)$, and $\bar{w} = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(w)$. Then, due to the assumed $H_1 \searrow\!/\!\curvearrowright_{C,R} (H_1, G_1, L_1)$, there must be an $(\pi, e, \mathcal{A})$-counterexample $((\mathit{\Gamma}', (w', <', \lesssim')), \tau')$ with $(\mathit{\Gamma}', (w', <', \lesssim')) \in H_1$, $\lhd = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta') \uplus \delta')(<')$, $\mathrel{\underset{\sim}{\lhd}} = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta') \uplus \delta')(\lesssim')$, and $\bar{w}' \lhd^+ \bar{w}$ for $\delta' := \epsilon(\pi)(\tau') \uplus \tau'$ and $\bar{w}' := \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta') \uplus \delta')(w')$. Thus, since $\bar{w}$ is $\lhd^+$-minimal in A, $((\mathit{\Gamma}', (w', <', \lesssim')), \tau') \notin D$. Thus, there must be some $(\pi, e, \mathcal{A})$-counterexample $((\mathit{\Gamma}'', (w'', <'', \lesssim'')), \tau'')$ either with $(\mathit{\Gamma}'', (w'', <'', \lesssim'')) \in L_1$, or otherwise with $(\mathit{\Gamma}'', (w'', <'', \lesssim'')) \in G_1$, $\lhd = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta'') \uplus \delta'')(<'')$, $\mathrel{\underset{\sim}{\lhd}} = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta'') \uplus \delta'')(\lesssim'')$, and $\bar{w}''\ (\mathrel{\underset{\sim}{\lhd}} \cup \lhd)^*\ \bar{w}'$ for $\delta'' := \epsilon(\pi)(\tau'') \uplus \tau''$ and $\bar{w}'' := \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta'') \uplus \delta'')(w'')$. In both cases, this contradicts $((\mathit{\Gamma}, (w, <, \lesssim)), \tau) \in D$; in the latter due to $\bar{w}''\ (\mathrel{\underset{\sim}{\lhd}} \cup \lhd)^*\ \bar{w}$. **Q.e.d. (Lemma 12.3)**

Proof of Theor. 13.6

Let $\mathcal{A} \in \mathrm{K}$ be arbitrary. Since $\mathcal{AX}$ is $\mathrm{V}_\exists{\times}\mathrm{V}_\forall$-valid in $\mathcal{A}$ (cf. Definition 13.2) and C is an R-choice-condition, $\mathcal{AX}$ is (C, R)-strongly valid in $\mathcal{A}$ by Lemma 10.2. By definition, this means that there is some existential $(\mathcal{A}, R)$-valuation e and some π that is $(e, \mathcal{A})$-compatible with (C, R) s.t. $\mathcal{AX}$ is strongly $(\pi, e, \mathcal{A})$-valid.

Claim 1: For i' with $i'\,(L{\cup}H)^*\,i$ and for $(i',((\Gamma',\aleph'),t'))\in F$: Γ' is strongly $(\pi,e,\mathcal{A})$-valid.
Proof of Claim 1: By induction on i' in $(L{\circ}H^*)^+$: Set $I := H^*\langle\!\{i'\}\!\rangle$. Due to $I \subseteq (L{\cup}H)^*\langle\!\{i\}\!\rangle$ and by the closedness assumption of the theorem we have $\mathrm{logic}(\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F))) \subseteq \mathrm{logic}(\mathrm{Goals}(\mathrm{Trees}(\langle (L{\cup}H)^*\langle\!\{i\}\!\rangle\rangle F))) \subseteq \mathcal{AX}$. Thus, $\mathrm{logic}(\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F)))$ is strongly $(\pi,e,\mathcal{A})$-valid. By induction hypothesis, $\mathrm{logic}(\mathrm{Hyps}(\langle L\langle I\rangle\rangle F))$ is strongly $(\pi,e,\mathcal{A})$-valid. Together this means that $\mathrm{logic}(\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F)) \cup \mathrm{Hyps}(\langle L\langle I\rangle\rangle F))$ is strongly $(\pi,e,\mathcal{A})$-valid, too. (Note that the last step would not be possible for (C,R)-strong validity instead of strong $(\pi,e,\mathcal{A})$-validity.) Since $(i',((\Gamma',\aleph'),t'))\in F$ and (F,C,R,L,H) satisfies the soundness invariant condition, $\{(\Gamma',\aleph')\} \curvearrowright_{C,R} (\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F)), \mathrm{Hyps}(\langle L\langle I\rangle\rangle F))$. All in all, by Lemma 12.3(1b), Γ' is strongly $(\pi,e,\mathcal{A})$-valid. Q.e.d. (Claim 1)

For the special case of $i'=i$, Claim 1 says that Γ is (C,R)-strongly valid in $\mathcal{A}$. Finally, by Lemma 10.3, $\Gamma\varsigma$ is $(\emptyset,R')$-strongly valid and R'-valid in $\mathcal{A}$. **Q.e.d. (Theor. 13.6)**

Proof of Theor. 13.7

$\emptyset$ is an $\emptyset$-choice-condition and $(\emptyset,\emptyset,\emptyset,\emptyset,\emptyset)$ vacuously satisfies the soundness invariant condition.

For the iteration steps, we let $(i'',((\Gamma'',\aleph''),t''))\in F'$ be arbitrary. Assuming the soundness invariant condition for (F,C,R,L,H) and using the abbreviations

$$\begin{array}{rcl|rcl} I & := & H^*\langle\!\{i''\}\!\rangle & I' & := & H'^*\langle\!\{i''\}\!\rangle \\ A & := & \mathrm{Goals}(\mathrm{Trees}(\langle I'\rangle F)) & A' & := & \mathrm{Goals}(\mathrm{Trees}(\langle I'\rangle F')) \\ & & & B' & := & \mathrm{Hyps}(\langle L'\langle I'\rangle\rangle F'). \end{array}$$

we have to show that C' is an R'-choice-condition and that $\{(\Gamma'',\aleph'')\}\curvearrowright_{C',R'}(A',B')$.

Hypothesizing: Note that F' is a partial function on $\mathbf{N}_+$ just like F because of $i\in\mathbf{N}_+\setminus\mathrm{dom}(F)$. Note that R is a variable-condition and that R^+ is a wellfounded ordering because C is an R-choice-condition (because (F,C,R,L,H) is assumed to be a proof forest).

$i''\in\mathrm{dom}(F)$: By assumption,

$$\{(\Gamma'',\aleph'')\} \curvearrowright_{C,R} (\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F)), \mathrm{Hyps}(\langle L\langle I\rangle\rangle F)).$$

As (C',R') is an extension of (C,R) and by Lemma 12.3(5), this means

$$\{(\Gamma'',\aleph'')\} \curvearrowright_{C',R'} (\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F)), \mathrm{Hyps}(\langle L\langle I\rangle\rangle F)).$$

Due to $H=H'$ we have $I=I'$, and then due to $L=L'$ and $F{\subseteq}F'$, we have

$$\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F)) \subseteq A' \quad\text{and}\quad \mathrm{Hyps}(\langle L\langle I\rangle\rangle F) \subseteq B'.$$

Thus, by Lemma 12.3(2), we have

$$\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F))\curvearrowright_{C',R'}(A',\emptyset) \quad\text{and}\quad \mathrm{Hyps}(\langle L\langle I\rangle\rangle F)\curvearrowright_{C',R'}(\emptyset,B').$$

Thus, by Lemma 12.3(3a,b), we have $\{(\Gamma'',\aleph'')\} \curvearrowright_{C',R'} (A',B')$.

$i''=i$: Then $\{(\Gamma'',\aleph'')\} = \{(\Gamma,\aleph)\} = \mathrm{Goals}(\{t\}) = \mathrm{Goals}(\{t''\}) \subseteq A' \subseteq A'\cup B'$. Thus, by Lemma 12.3(2), $\{(\Gamma'',\aleph'')\} \curvearrowright_{C',R'} (A',B')$.

Expansion: Note that $\forall J.\ \mathrm{Hyps}(\langle J\rangle F) = \mathrm{Hyps}(\langle J\rangle F')$.

Claim 1: $\mathrm{Hyps}(\langle I'\rangle F)\curvearrowright_{C',R'}(A,B')$.

Claim 2: $A \searrow\!/\!\curvearrowright_{C',R'} (\mathrm{Hyps}(\langle I'\rangle F), A', B')$.

By Claim 1, Claim 2, and Lemma 12.3(3a), we get

$$\mathrm{Hyps}(\langle I'\rangle F) \searrow\!/\!\curvearrowright_{C',R'} (\mathrm{Hyps}(\langle I'\rangle F), A', B').$$

By Lemma 12.3(7), we get $\mathrm{Hyps}(\langle I'\rangle F)\curvearrowright_{C',R'}(A',B')$. Since $\{(\Gamma'',\aleph'')\} \subseteq \mathrm{Hyps}(\langle I'\rangle F)$, we have $\{(\Gamma'',\aleph'')\}\curvearrowright_{C',R'}(\mathrm{Hyps}(\langle I'\rangle F),\emptyset)$ by Lemma 12.3(2). Thus, by Lemma 12.3(3a), we get $\{(\Gamma'',\aleph'')\}\curvearrowright_{C',R'}(A',B')$.

Proof of Claim 1: By Lemma 12.3(4) it suffices to show $\mathrm{Hyps}(\langle\{i'''\}\rangle F)\curvearrowright_{C',R'}(A,B')$ for any $i'''\in I'$. We have

$$\mathrm{Hyps}(\langle\{i'''\}\rangle F)\ \curvearrowright_{C,R}\ (\mathrm{Goals}(\mathrm{Trees}(\langle I'''\rangle F)),\mathrm{Hyps}(\langle L\langle I'''\rangle\rangle F))$$

for $I''':=H^*\langle\{i'''\}\rangle$ by assumption. As (C',R') is an extension of (C,R) and by Lemma 12.3(5),

$$\mathrm{Hyps}(\langle\{i'''\}\rangle F)\ \curvearrowright_{C',R'}\ (\mathrm{Goals}(\mathrm{Trees}(\langle I'''\rangle F)),\mathrm{Hyps}(\langle L\langle I'''\rangle\rangle F)).$$

Due to $H\subseteq H'$, we have $I'''\subseteq H'^*\langle\{i'''\}\rangle\subseteq H'^*\langle I'\rangle=I'$. Thus, $\mathrm{Goals}(\mathrm{Trees}(\langle I'''\rangle F))\subseteq A$ and (due to $L\subseteq L'$) $\mathrm{Hyps}(\langle L\langle I'''\rangle\rangle F)\subseteq\mathrm{Hyps}(\langle L'\langle I'\rangle\rangle F)=B'$. Thus, by Lemma 12.3(2), we get $\mathrm{Goals}(\mathrm{Trees}(\langle I'''\rangle F))\curvearrowright_{C',R'}(A,\emptyset)$ and $\mathrm{Hyps}(\langle L\langle I'''\rangle\rangle F)\curvearrowright_{C',R'}(\emptyset,B')$. By Lemma 12.3(3a,b): $\mathrm{Hyps}(\langle\{i'''\}\rangle F)\curvearrowright_{C',R'}(A,B')$. Q.e.d. (Claim 1)

Proof of Claim 2: If $i\notin I'$, then we have $A=A'$ and Claim 2 follows from Lemma 12.3(2). Thus, we may assume $i\in I'$. By construction of t' we have $A\setminus\{(\Delta,\sqsupset)\}\subseteq A'$. Thus, by Lemma 12.3(2),

$$A\setminus\{(\Delta,\sqsupset)\}\searrow\!/\!\curvearrowright_{C',R'}(\mathrm{Hyps}(\langle I'\rangle F),A',B').$$

By assumption we have

$$\{(\Delta,\sqsupset)\}\searrow\!/\!\curvearrowright_{C',R'}(\mathrm{Hyps}(\langle N_{\mathrm{H}}\rangle F),G,\mathrm{Hyps}(\langle N_{\mathrm{L}}\rangle F)).$$

By Lemma 12.3(4), we get Claim 2 due to $\mathrm{Hyps}(\langle N_{\mathrm{H}}\rangle F)\subseteq\mathrm{Hyps}(\langle I'\rangle F)$, $G\subseteq\mathrm{Goals}(\{t'\})=\mathrm{Goals}(\mathrm{Trees}(\langle\{i\}\rangle F'))\subseteq A'$, and $\mathrm{Hyps}(\langle N_{\mathrm{L}}\rangle F)\subseteq\mathrm{Hyps}(\langle L'\langle I'\rangle\rangle F)=B'$, which hold due to $N_{\mathrm{H}}\subseteq H'\langle\{i\}\rangle\subseteq H'\langle I'\rangle=I'$, the construction of t', and $N_{\mathrm{L}}\subseteq L'\langle\{i\}\rangle\subseteq L'\langle I'\rangle$, resp.. Q.e.d. (Claim 2)

Instantiation: By Lemma 9.6, C' is an R'-choice-condition.

Set $O:=D(i'')$ and $N:=\mathrm{dom}(C)\cap\langle(\mathrm{dom}(C)\cap\mathrm{dom}(\sigma))\setminus O\rangle R^*$.

Claim 3: $O\subseteq\mathrm{dom}(C)\cap\mathrm{dom}(\sigma)\subseteq O\uplus N$, $\mathrm{dom}(C)\cap\langle N\rangle R^+\subseteq N$, $N\subseteq\mathrm{dom}(C)\setminus O$, and $N\cap\mathcal{V}(\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F)),\mathrm{Hyps}(\langle\{i''\}\cup L\langle I\rangle\rangle F))=\emptyset$.

Proof of Claim 3: By definition of $D(i)$ and N, the first, second, and third statement are trivial with the exception of $N\cap O=\emptyset$, which we will show together with the last statement: Set $M:=R^*\langle\mathcal{V}_{\forall,\mathrm{s}}(\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F)),\mathrm{Hyps}(\langle\{i''\}\cup L\langle I\rangle\rangle F))\rangle$. It now suffices to show $N\cap M=\emptyset$. If $z_1^{\forall,\mathrm{s}}\in N$, there is some $z_0^{\forall,\mathrm{s}}\in(\mathrm{dom}(C)\cap\mathrm{dom}(\sigma))\setminus O$ with $z_0^{\forall,\mathrm{s}}\ R^*\ z_1^{\forall,\mathrm{s}}$, but then, if $z_1^{\forall,\mathrm{s}}\in M$, we get $z_0^{\forall,\mathrm{s}}\in M$ and the contradictory $z_0^{\forall,\mathrm{s}}\in O$ by definition of O. Q.e.d. (Claim 3)

By assumption $\mathrm{Hyps}(\langle\{i''\}\rangle F)\ \curvearrowright_{C,R}\ (\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F)),\mathrm{Hyps}(\langle L\langle I\rangle\rangle F))$. Set $B'':=\bigcup_{y^{\forall,\mathrm{s}}\in O}\mathrm{Hyps}(\langle\{j_{y^{\forall,\mathrm{s}}}\}\rangle F')$. Then we have $\mathrm{logic}(B'')=\langle O\rangle Q_{C,\sigma}$ according to the requirements of the Instantiation rule. By Claim 3 we can apply Lemma 12.3(6) to get:

$$\begin{array}{rcl}\{(\Gamma'',\aleph'')\}=\mathrm{Hyps}(\langle\{i''\}\rangle F)\sigma & \curvearrowright_{C',R'} & (\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F))\sigma,\mathrm{Hyps}(\langle L\langle I\rangle\rangle F)\sigma\cup B'')\\ & = & (\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F')),\mathrm{Hyps}(\langle L\langle I\rangle\rangle F')\cup B'')\\ & = & (A',\mathrm{Hyps}(\langle L\langle I'\rangle\rangle F')\cup B''),\end{array}$$

the latter step being due to $I=I'$.

By definition of L' we have $\{\ j_{y^{\forall,\mathrm{s}}}\mid y^{\forall,\mathrm{s}}\in O\ \}\subseteq L'\langle\{i''\}\rangle\subseteq L'\langle I'\rangle$. Thus, we have $B''\subseteq B'$. Moreover, due to $L\subseteq L'$, we have $\mathrm{Hyps}(\langle L\langle I'\rangle\rangle F')\subseteq B'$. Together this implies $\mathrm{Hyps}(\langle L\langle I'\rangle\rangle F')\cup B''\ \curvearrowright_{C',R'}\ (\emptyset,B')$, by Lemma 12.3(2). By Lemma 12.3(3b) we get $\{(\Gamma'',\aleph'')\}\ \curvearrowright_{C',R'}\ (A',B')$. **Q.e.d. (Theor. 13.7)**

Proof of Theor. 13.10

The empty proof forest trivially satisfies the safeness invariant condition.

Hypothesizing: When we assume the old trees from F to satisfy the safeness invariant condition for (C,R), then they also satisfy it for (C',R') by Lemma 11.2(5b) because (C',R') is an extension of (C,R). The new tree $(i,((\Gamma,\aleph),t))$ satisfies the safeness invariant condition because $\mathrm{logic}(\mathrm{Goals}(\{t\}))=\{\Gamma\}$ and $\{\Gamma\}$ (C',R')-reduces to $\{\Gamma\}$ by Lemma 11.2(2).

Expansion: When we assume the non-expanded trees to satisfy the safeness invariant condition for (C,R), then they also satisfy it for the extension (C',R') of (C,R) by Lemma 11.2(5b). For the new tree $(i,(\Gamma,t'))$ we have to show that $\mathrm{logic}(\mathrm{Goals}(\{t'\}))$ (C',R')-reduces to $\{\Gamma\}$.
Claim 1: $\mathrm{logic}(G)$ (C',R')-reduces to $\{\Delta\}$.
Proof of Claim 1: In case of a sequent calculus this is given by the additional requirement of safeness of the Expansion step. In case of a tableau calculus we have $\mathrm{logic}(G)=\{\ \Pi\Delta \mid \Pi\in M\ \}$, and the claim follows because because strong $(\pi,e,\mathcal{A})$-validity of Δ implies strong $(\pi,e,\mathcal{A})$-validity of $\Pi\Delta$. Q.e.d. (Claim 1)
Claim 2: $\mathrm{logic}(\mathrm{Goals}(\{t'\}))$ (C',R')-reduces to $\mathrm{logic}(\mathrm{Goals}(\{t\}))$.
Proof of Claim 2: As $\mathrm{Goals}(\{t'\})\setminus G\subseteq\mathrm{Goals}(\{t\})$,

$$\mathrm{logic}(\mathrm{Goals}(\{t'\}))\setminus\mathrm{logic}(G)\ \ (C',R')\text{-reduces to }\mathrm{logic}(\mathrm{Goals}(\{t\}))$$

by Lemma 11.2(2). Thus, by Claim 1, the claim follows by Lemma 11.2(4) due to $\Delta\in\mathrm{logic}(\mathrm{Goals}(\{t\}))$. Q.e.d. (Claim 2)

When we assume the old tree $(i,((\Gamma,\aleph),t))$ to satisfy the safeness invariant condition for (C,R), then $\mathrm{logic}(\mathrm{Goals}(\{t\}))$ (C',R')-reduces to $\{\Gamma\}$ by Lemma 11.2(5b). By Lemma 11.2(3), together with Claim 2 this implies that $\mathrm{logic}(\mathrm{Goals}(\{t'\}))$ (C',R')-reduces to $\{\Gamma\}$, as was to be shown.

Instantiation: Assume any old tree $(i,((\Gamma,\aleph),t))\in F$ to satisfy the safeness invariant condition for (C,R), i.e. $\mathrm{logic}(\mathrm{Goals}(\{t\}))$ (C,R)-reduces to $\{\Gamma\}$. Set $O:=D(i)$ and $N:=\mathrm{dom}(C)\cap\langle(\mathrm{dom}(C)\cap\mathrm{dom}(\sigma))\setminus O\rangle R^*$.
Claim 3: $O\subseteq\mathrm{dom}(C)\cap\mathrm{dom}(\sigma)\subseteq O\uplus N$, $\mathrm{dom}(C)\cap\langle N\rangle R^+\subseteq N$, $N\subseteq\mathrm{dom}(C)\setminus O$, and $N\cap\mathcal{V}(\mathrm{Goals}(\mathrm{Trees}(\langle I\rangle F)),\mathrm{Hyps}(\langle\{i\}\cup L\langle I\rangle\rangle F))=\emptyset$.
Proof of Claim 3: Just like the proof of Claim 3 in the proof of Theor. 13.7. Q.e.d. (Claim 3)

By Lemma 11.2(6b) and Claim 3, $\mathrm{logic}(\mathrm{Goals}(\{t\sigma\}))$ (C',R')-reduces to $\{\Gamma\sigma\}\cup\langle O\rangle Q_{C,\sigma}$. As the Instantiation step is safe by assumption, by Theor. 13.7 and Theor. 13.6, $\langle O\rangle Q_{C,\sigma}$ is (C',R')-strongly valid. Thus, $\mathrm{logic}(\mathrm{Goals}(\{t\sigma\}))$ (C',R')-reduces to $\{\Gamma\sigma\}$, as was to be shown. **Q.e.d. (Theor. 13.10)**

Proof of Theor. 14.1

We only prove one example of each kind of rule to be a safe sub-rule of the Expansion rule and leave the others as an exercise.

Due to $\mathrm{ran}(G)=\{\sqsupset\}$, for the α-, β-, γ-, Rewrite-, and Cut-rules, it suffices to show that, for each Σ-structure $\mathcal{A}$, each existential $(\mathcal{A},R)$-valuation e, each π that is $(e,\mathcal{A})$-compatible with (C,R), each $\tau\in\mathrm{V}_{\forall,\mathrm{w}}\rightarrow\mathcal{A}$, and for $\delta:=\epsilon(\pi)(\tau)\uplus\tau$, the $(\delta,e,\mathcal{A})$-validity of $\{\Delta\}$ is logically equivalent to that of $\mathrm{logic}(G)$.

<u>α-rule:</u> $(\delta,e,\mathcal{A})$-validity of $\{\varGamma\ \ (A{\vee}B)\ \ \varPi\}$ is indeed logically equivalent to that of $\{A\ B\ \varGamma\ \varPi\}$.

<u>β-rule:</u> $(\delta,e,\mathcal{A})$-validity of $\{\varGamma\ \ (A{\wedge}B)\ \ \varPi\}$ is indeed logically equivalent to that of $\{A\ \varGamma\ \varPi,\qquad B\ [\,\overline{A}\,]\ \varGamma\ \varPi\}$.

<u>γ-rule:</u> $(\delta,e,\mathcal{A})$-validity of $\{\varGamma\ \ \exists x.\ A\ \ \varPi\}$ is indeed logically equivalent to that of $\{A\{x{\mapsto}x^{\exists}\}\ \ \varGamma\ \ \exists x.\ A\ \ \varPi\}$. The implication from left to right is given because the former sequent is a sub-sequent of the latter. For the other direction, note that although it is clear that $(\delta,e,\mathcal{A})$-validity of $A\{x{\mapsto}x^{\exists}\}$ implies $(\delta,e,\mathcal{A})$-validity of $\exists x.\ A$, we should be a little more explicit here because the standard semantical definition of $\exists$ (cf. e.g. Wirth (1997), p. 188, or Enderton (1973), p. 82) uses a free universal variable instead of a free existential variable and is somewhat more complicated than it could be in terms of free existential variables. Moreover, in the note above the rule we remarked that the restriction on $x^{\exists}$ is not really necessary. Thus, in order to be more explicit here, assume that $A\{x{\mapsto}x^{\exists}\}$ is $(\delta,e,\mathcal{A})$-valid. Let $y^{\forall}\in\mathrm{V}_{\forall}\backslash\mathcal{V}(A)$. Then, since $A\{x{\mapsto}y^{\forall}\}\{y^{\forall}{\mapsto}x^{\exists}\}$ is equal to $A\{x{\mapsto}x^{\exists}\}$, we know that $A\{x{\mapsto}y^{\forall}\}\{y^{\forall}{\mapsto}x^{\exists}\}$ is valid in $\mathcal{A}\uplus\epsilon(e)(\delta)\uplus\delta$. Then, by the Substitution-Lemma, $A\{x{\mapsto}y^{\forall}\}$ is valid in $\mathcal{A}\uplus\epsilon(e)(\delta)\uplus\delta'$ for $\delta'\in\mathrm{V}_{\forall}\rightarrow\mathcal{A}$ given by ${}_{\mathrm{V}_{\forall}\backslash\{y^{\forall}\}}{\upharpoonright}\delta':={}_{\mathrm{V}_{\forall}\backslash\{y^{\forall}\}}{\upharpoonright}\delta$ and $\delta'(y^{\forall}):=\epsilon(e)(\delta)(x^{\exists})$. By the standard semantical definition of $\exists$ and since binding of x cannot occur in A (as $\exists x.\ A$ is a formula in our restricted sense, cf. Section 3.1), this means that $\exists x.\ (A\{x{\mapsto}y^{\forall}\}\{y^{\forall}{\mapsto}x\})$ is valid in $\mathcal{A}\uplus\epsilon(e)(\delta)\uplus\delta$. Since $y^{\forall}$ does not occur in A, this formula is equal to $\exists x.\ A$, which means that the former sequent is $(\delta,e,\mathcal{A})$-valid.

<u>Rewrite-rule:</u> We have to show that $(\delta,e,\mathcal{A})$-validity of $\{\varGamma\ \ A[s]\ \ \varPi\ \ B\ \ \varLambda\}$ is logically equivalent to that of $\{A[t]\ \ \varGamma\ \ \varPi\ \ B\ \ \varLambda\}$.

If $\mathrm{eval}(\mathcal{A}\uplus\epsilon(e)(\delta)\uplus\delta)(s)\neq\mathrm{eval}(\mathcal{A}\uplus\epsilon(e)(\delta)\uplus\delta)(t)$, then both are $(\delta,e,\mathcal{A})$-valid because B is.

Otherwise, we set $a:=\mathrm{eval}(\mathcal{A}\uplus\epsilon(e)(\delta)\uplus\delta)(s)$, choose some $z^{\forall}\in\mathrm{V}_{\forall}\backslash\mathcal{V}(A[s])$, and define $\delta'\in\mathrm{V}_{\forall}\rightarrow\mathcal{A}$ by ${}_{\mathrm{V}_{\forall}\backslash\{z^{\forall}\}}{\upharpoonright}\delta':={}_{\mathrm{V}_{\forall}\backslash\{z^{\forall}\}}{\upharpoonright}\delta$ and $\delta'(z^{\forall}):=a$. Then $a=\mathrm{eval}(\mathcal{A}\uplus\epsilon(e)(\delta)\uplus\delta)(t)$. Moreover, by the Substitution-Lemma:

$$\begin{array}{ll}
\mathrm{eval}(\mathcal{A}\uplus\epsilon(e)(\delta)\uplus\delta)(A[s]) & =\\
\mathrm{eval}(\mathcal{A}\uplus\epsilon(e)(\delta)\uplus\delta)(A[z^{\forall}]\{z^{\forall}{\mapsto}s\}) & =\\
\mathrm{eval}(\mathcal{A}\uplus\epsilon(e)(\delta)\uplus\delta')(A[z^{\forall}]) & =\\
\mathrm{eval}(\mathcal{A}\uplus\epsilon(e)(\delta)\uplus\delta)(A[z^{\forall}]\{z^{\forall}{\mapsto}t\}) & =\\
\mathrm{eval}(\mathcal{A}\uplus\epsilon(e)(\delta)\uplus\delta)(A[t]). &
\end{array}$$

Note that the usual problems with variables getting captured by binders cannot occur in our context, because the unbound occurrence of variables from $\mathrm{V}_{\mathrm{bound}}$ in formulas (like $(s{\neq}t)$) is not permitted, cf. Section 3.1.

<u>Cut:</u> Trivial.

δ-rule: Note that in this proof, we only make use of the weaker conditions on the occurrence of $x^{\forall,\mathrm{w}}$ given by Note 4.

Claim 1: (C', R') is an extension of (C, R).
Proof of Claim 1: Since (F, C, R, L, H) is a proof forest, C is an R-choice-condition. Moreover, $C \subseteq C'$ and $R \subseteq R'$ are trivial. Thus, we only have to show that C' is an R'-choice-condition. As $C' = C$, we only have to show that R'^{+} is a wellfounded ordering.
As $\mathrm{ran}(R'') = \{x^{\forall,\mathrm{w}}\}$ and $\{x^{\forall,\mathrm{w}}\} \cap \mathrm{dom}(R) = \emptyset$ is required, we have $R'' \circ R = \emptyset$. As $\mathrm{ran}(R'') \cap \mathrm{dom}(R'') \subseteq \mathrm{V}_{\forall,\mathrm{w}} \cap (\mathrm{V}_{\exists} \cup \mathrm{V}_{\forall,\mathrm{s}}) = \emptyset$, we have $R'' \circ R'' = \emptyset$. Therefore, as R^{+} is a wellfounded ordering, R'^{+} is a wellfounded ordering, too. Q.e.d. (Claim 1)

Now, we have to show that

$$\{(\Gamma\ \ \forall x.\ A\ \ \Pi,\ \sqsupseteq)\} \curvearrowright_{C',R'} (\{(A\{x \mapsto x^{\forall,\mathrm{w}}\}\ \Gamma\ \Pi,\ \sqsupseteq)\}, \emptyset)$$

Let e and π be arbitrary s.t. e is an existential $(\mathcal{A}, R')$-valuation and π is $(e, \mathcal{A})$-compatible with (C', R'). Assume that $((\Gamma\ \ \forall x.\ A\ \ \Pi,\ \sqsupseteq), \tau)$ is an $(\pi, e, \mathcal{A})$-counterexample. Then, $\Gamma\Pi$ is invalid in $\mathcal{A} \uplus \epsilon(e)(\epsilon(\pi)(\tau) \uplus \tau) \uplus \epsilon(\pi)(\tau) \uplus \tau$.

Claim 2: $\Gamma\Pi$ is invalid in $\mathcal{A} \uplus \epsilon(e)(\epsilon(\pi)(\tau') \uplus \tau') \uplus \epsilon(\pi)(\tau') \uplus \tau'$ and

$$\begin{aligned} &\mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\epsilon(\pi)(\tau') \uplus \tau') \uplus \epsilon(\pi)(\tau') \uplus \tau')(\sqsupseteq) \\ =\ &\mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\epsilon(\pi)(\tau) \uplus \tau) \uplus \epsilon(\pi)(\tau) \uplus \tau)(\sqsupseteq) \end{aligned}$$

for all $\tau' \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$ with ${}_{\mathrm{V}_{\forall,\mathrm{w}} \setminus \{x^{\forall,\mathrm{w}}\}}{\upharpoonleft}\tau' = {}_{\mathrm{V}_{\forall,\mathrm{w}} \setminus \{x^{\forall,\mathrm{w}}\}}{\upharpoonleft}\tau$.
Proof of Claim 2: Otherwise there must be some $u \in \mathcal{V}_{\forall,\mathrm{s}}(\Gamma\Pi, \sqsupseteq) \cup \mathcal{V}_{\exists}(\Gamma\Pi, \sqsupseteq)$ with $x^{\forall,\mathrm{w}}\ S_\pi \circ S_e\ u$ (the first occurrence of τ' makes a difference) or $x^{\forall,\mathrm{w}}\ S_e\ u$ (the second occurrence of τ' makes a difference) or $x^{\forall,\mathrm{w}}\ S_\pi\ u$ when the third occurrence of τ' makes a difference. Note that the fourth occurrence of τ' cannot make a difference simply because $x^{\forall,\mathrm{w}}$ does not occur in $\mathcal{V}(\Gamma\Pi, \sqsupseteq)$. Since $u\ R''\ x^{\forall,\mathrm{w}}$, we know that $(R' \cup S_e \cup S_\pi)^{+}$ is not a wellfounded ordering, which contradicts π being $(e, \mathcal{A})$-compatible with (C', R'). Q.e.d. (Claim 2)

Now, if there is any $\tau' \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$ with ${}_{\mathrm{V}_{\forall,\mathrm{w}} \setminus \{x^{\forall,\mathrm{w}}\}}{\upharpoonleft}\tau' = {}_{\mathrm{V}_{\forall,\mathrm{w}} \setminus \{x^{\forall,\mathrm{w}}\}}{\upharpoonleft}\tau$ s.t. $A\{x \mapsto x^{\forall,\mathrm{w}}\}$ is invalid in $\mathcal{A} \uplus \epsilon(e)(\epsilon(\pi)(\tau') \uplus \tau') \uplus \epsilon(\pi)(\tau') \uplus \tau'$, then due to Claim 2 $((A\{x \mapsto x^{\forall,\mathrm{w}}\}\ \Gamma\ \Pi,\ \sqsupseteq), \tau')$ is the $(\pi, e, \mathcal{A})$-counterexample we are searching for. Otherwise, $A\{x \mapsto x^{\forall,\mathrm{w}}\}$ is valid in $\mathcal{A} \uplus \epsilon(e)(\epsilon(\pi)(\tau') \uplus \tau') \uplus \epsilon(\pi)(\tau') \uplus \tau'$ for all $\tau' \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$ with ${}_{\mathrm{V}_{\forall,\mathrm{w}} \setminus \{x^{\forall,\mathrm{w}}\}}{\upharpoonleft}\tau' = {}_{\mathrm{V}_{\forall,\mathrm{w}} \setminus \{x^{\forall,\mathrm{w}}\}}{\upharpoonleft}\tau$.

Claim 4: $A\{x \mapsto x^{\forall,\mathrm{w}}\}$ is valid in $\mathcal{A} \uplus \epsilon(e)(\epsilon(\pi)(\tau) \uplus \tau) \uplus \epsilon(\pi)(\tau) \uplus \tau'$ for all $\tau' \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$ with ${}_{\mathrm{V}_{\forall,\mathrm{w}} \setminus \{x^{\forall,\mathrm{w}}\}}{\upharpoonleft}\tau' = {}_{\mathrm{V}_{\forall,\mathrm{w}} \setminus \{x^{\forall,\mathrm{w}}\}}{\upharpoonleft}\tau$.

Proof of Claim 4: Otherwise there must be some $u \in \mathcal{V}_{\forall,\mathrm{s}}(A\{x \mapsto x^{\forall,\mathrm{w}}\}) \cup \mathcal{V}_{\exists}(A\{x \mapsto x^{\forall,\mathrm{w}}\})$ with $x^{\forall,\mathrm{w}}\ S_\pi \circ S_e\ u$ (the first occurrence of τ makes a difference) or $x^{\forall,\mathrm{w}}\ S_e\ u$ (the second occurrence of τ makes a difference) or $x^{\forall,\mathrm{w}}\ S_\pi\ u$ when the third occurrence of τ makes a difference. Since $u\ R''\ x^{\forall,\mathrm{w}}$, we know that $(R' \cup S_e \cup S_\pi)^{+}$ is not a wellfounded ordering, which contradicts π being $(e, \mathcal{A})$-compatible with (C', R'). Q.e.d. (Claim 4)

By the standard semantical definition of $\forall$ (cf. e.g. Wirth (1997), p. 188, or Enderton (1973), p. 82) and since binding of x cannot occur in A (as $\forall x.\ A$ is a formula in our restricted sense, cf. Section 3.1), Claim 4 means that $\forall x.\ (A\{x \mapsto x^{\forall,\mathrm{w}}\}\{x^{\forall,\mathrm{w}} \mapsto x\})$ is valid in $\mathcal{A} \uplus \epsilon(e)(\epsilon(\pi)(\tau) \uplus \tau) \uplus \epsilon(\pi)(\tau) \uplus \tau$, i.e. $(\epsilon(\pi)(\tau) \uplus \tau, e, \mathcal{A})$-valid. Since $x^{\forall,\mathrm{w}}$ does not occur in A, this formula is equal to $\forall x.\ A$, which contradicts $((\Gamma\ \ \forall x.\ A\ \ \Pi,\ \sqsupseteq), \tau)$ being an $(\pi, e, \mathcal{A})$-counterexample.

For the safeness proof, assume that $\Gamma\ \ \forall x.\ A\ \ \Pi$ is strongly $(\pi, e, \mathcal{A})$-valid. For arbitrary $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$ we have to show that $A\{x \mapsto x^{\forall,\mathrm{w}}\}\ \Gamma\ \Pi$ is $(\delta, e, \mathcal{A})$-valid for $\delta := \epsilon(\pi)(\tau) \uplus \tau$. If some formula in $\Gamma\Pi$ is $(\delta, e, \mathcal{A})$-valid, then the latter sequent is $(\delta, e, \mathcal{A})$-valid, too. Otherwise, $\forall x.\ A$ is $(\delta, e, \mathcal{A})$-valid. Then, by the standard semantical definition of $\forall$, $A\{x \mapsto x^{\forall,\mathrm{w}}\}$ is $(\delta, e, \mathcal{A})$-valid, too, as was to be shown.

Liberalized δ-rule: Note that in this proof, we only make use of the weaker conditions on the occurrence of $x^{\forall,\mathrm{s}}$ given by Note 5.

Claim 5: (C', R') is an extension of (C, R).

Proof of Claim 5: Since (F, C, R, L, H) is a proof forest, C is an R-choice-condition. Moreover, $C \subseteq C'$ and $R \subseteq R'$ are trivial. Thus, we only have to show that C' is an R'-choice-condition.

As $x^{\forall,\mathrm{s}} \in \mathrm{V}_{\forall,\mathrm{s}} \setminus \mathrm{dom}(C)$ is required, C' is also a partial function on $\mathrm{V}_{\forall,\mathrm{s}}$.

As $\mathrm{ran}(R'') = \{x^{\forall,\mathrm{s}}\}$ and $\{x^{\forall,\mathrm{s}}\} \cap \mathrm{dom}(R) = \emptyset$ is required, we have $R'' \circ R = \emptyset$. As $\mathrm{ran}(R'') \cap \mathrm{dom}(R'') = \{x^{\forall,\mathrm{s}}\} \cap \mathcal{V}_{\mathrm{free}}(A) = \{x^{\forall,\mathrm{s}}\} \cap \mathcal{V}(A) = \emptyset$ is required, we have $R'' \circ R'' = \emptyset$. Therefore, as R^+ is a wellfounded ordering, R'^+ is a wellfounded ordering, too.

Moreover, for $y^{\forall,\mathrm{s}} \in \mathrm{dom}(C')$, we either have $y^{\forall,\mathrm{s}} \in \mathrm{dom}(C)$ and then $(\mathcal{V}_{\mathrm{free}}(C'(y^{\forall,\mathrm{s}})) \setminus \{y^{\forall,\mathrm{s}}\}) \times \{y^{\forall,\mathrm{s}}\} = (\mathcal{V}_{\mathrm{free}}(C(y^{\forall,\mathrm{s}})) \setminus \{y^{\forall,\mathrm{s}}\}) \times \{y^{\forall,\mathrm{s}}\} \subseteq R^+ \subseteq R'^+$, or $y^{\forall,\mathrm{s}} = x^{\forall,\mathrm{s}}$ and then $(\mathcal{V}_{\mathrm{free}}(C'(y^{\forall,\mathrm{s}})) \setminus \{y^{\forall,\mathrm{s}}\}) \times \{y^{\forall,\mathrm{s}}\} = (\mathcal{V}_{\mathrm{free}}(A\{x \mapsto x^{\forall,\mathrm{s}}\}) \setminus \{x^{\forall,\mathrm{s}}\}) \times \{x^{\forall,\mathrm{s}}\} \subseteq \mathcal{V}_{\mathrm{free}}(A) \times \{x^{\forall,\mathrm{s}}\} = R'' \subseteq R'^+$. Q.e.d. (Claim 5)

Now, due to $\mathrm{ran}(G) = \{\sqsupset\}$, it suffices to show that, for each Σ-structure $\mathcal{A}$, each existential $(\mathcal{A}, R')$-valuation e, each π that is $(e, \mathcal{A})$-compatible with (C', R'), each $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$, and for $\delta := \epsilon(\pi)(\tau) \uplus \tau$, the $(\delta, e, \mathcal{A})$-validity of

$$\Gamma\ \ \forall x.\ A\ \ \Pi \text{ is logically equivalent to that of } A\{x \mapsto x^{\forall,\mathrm{s}}\}\ \Gamma\ \Pi.$$

For the soundness direction, we have to show that the former sequent is $(\delta, e, \mathcal{A})$-valid under the assumption that the latter is. If some formula in $\Gamma\Pi$ is $(\delta, e, \mathcal{A})$-valid, then the former sequent is $(\delta, e, \mathcal{A})$-valid, too. Otherwise, this means that $A\{x \mapsto x^{\forall,\mathrm{s}}\}$ is $(\delta, e, \mathcal{A})$-valid. Since π is $(e, \mathcal{A})$-compatible with (C', R'), by item 2 of Definition 9.2, we know that $A\{x \mapsto x^{\forall,\mathrm{s}}\}$ is $(\delta', e, \mathcal{A})$-valid for all $\delta' \in \mathrm{V}_\forall \to \mathcal{A}$ with ${}_{\mathrm{V}_\forall \setminus \{x^{\forall,\mathrm{s}}\}}{\upharpoonright}\delta' = {}_{\mathrm{V}_\forall \setminus \{x^{\forall,\mathrm{s}}\}}{\upharpoonright}\delta$. This means that $A\{x \mapsto x^{\forall,\mathrm{s}}\}$ is valid in $\mathcal{A} \uplus \epsilon(e)(\delta') \uplus \delta'$ for all $\delta' \in \mathrm{V}_\forall \to \mathcal{A}$ with ${}_{\mathrm{V}_\forall \setminus \{x^{\forall,\mathrm{s}}\}}{\upharpoonright}\delta' = {}_{\mathrm{V}_\forall \setminus \{x^{\forall,\mathrm{s}}\}}{\upharpoonright}\delta$.

Claim 6: $A\{x \mapsto x^{\forall,\mathrm{s}}\}$ is valid in $\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta'$ for all $\delta' \in \mathrm{V}_\forall \to \mathcal{A}$ with ${}_{\mathrm{V}_\forall \setminus \{x^{\forall,\mathrm{s}}\}}{\upharpoonright}\delta' = {}_{\mathrm{V}_\forall \setminus \{x^{\forall,\mathrm{s}}\}}{\upharpoonright}\delta$.

Proof of Claim 6: Otherwise we have $x^{\forall,\mathrm{s}} S_e u^\exists$ for some $u^\exists \in \mathcal{V}_\exists(A\{x \mapsto x^{\forall,\mathrm{s}}\})$. But then $u^\exists \in \mathcal{V}_{\mathrm{free}}(A)$ and then $u^\exists R'' x^{\forall,\mathrm{s}}$. This means that $(R' \cup S_e)^+$ is not a wellfounded ordering, which contradicts e being an existential $(\mathcal{A}, R')$-valuation. Q.e.d. (Claim 6)

By the standard semantical definition of $\forall$ (cf. e.g. Wirth (1997), p. 188, or Enderton (1973), p. 82) and since binding of x cannot occur in A (as $\forall x.\ A$ is a formula in our restricted sense, cf. Section 3.1), Claim 6 means that $\forall x.\ (A\{x \mapsto x^{\forall,\mathrm{s}}\}\{x^{\forall,\mathrm{s}} \mapsto x\})$ is valid in $\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta$. Since $x^{\forall,\mathrm{s}}$ does not occur in A, this formula is equal to $\forall x.\ A$, which means that the former sequent is $(\delta, e, \mathcal{A})$-valid as was to be shown.

For the safeness direction, we have to show that the latter sequent is $(\delta, e, \mathcal{A})$-valid under the assumption that the former is. If some formula in $\Gamma\Pi$ is $(\delta, e, \mathcal{A})$-valid, then the latter sequent is $(\delta, e, \mathcal{A})$-valid, too. Otherwise, $\forall x.\ A$ is $(\delta, e, \mathcal{A})$-valid. Then, by the standard semantical definition of $\forall$, $A\{x \mapsto x^{\forall,\mathrm{s}}\}$ is $(\delta, e, \mathcal{A})$-valid, too, as was to be shown.

Q.e.d. (Theor. 14.1)

Proof of Theor. 14.3

Let $G := \{\ (\Pi\Delta, \sqsupset)\ |\ \Pi \in M\ \}$ as in the Expansion rule of the tableau calculus of Definition 13.4. According to this definition of an Expansion step we have to show

$$\{(\Delta, \sqsupset)\} \searrow\!/\!\curvearrowright_{C',R'} (\{(\Phi, \urcorner)\}, G, \emptyset)$$

in case of "induction hypothesis application" and

$$\{(\Delta, \sqsupset)\} \searrow\!/\!\curvearrowright_{C',R'} (\emptyset, G, \{(\Phi, \urcorner)\})$$

in case of "lemma application". According to Definition 12.2 and due to $\mathrm{ran}(G) = \{\sqsupset\}$, it is sufficient to show that, for $\mathcal{A} \in \mathrm{K}$, e an existential $(\mathcal{A}, R')$-valuation, and π $(e, \mathcal{A})$-compatible with (C', R'), for any $(\pi, e, \mathcal{A})$-counterexample $((\Delta, \sqsupset), \tau)$, under the assumption that logic(G) is $(\delta, e, \mathcal{A})$-valid for $\delta := \epsilon(\pi)(\tau) \uplus \tau$, there is an $(\pi, e, \mathcal{A})$-counterexample $((\Phi, \urcorner), \tau')$ s.t. (in case of hypothesis application only), for $\lhd := \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(<)$, $\precapprox := \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(\lesssim)$, $\bar{w} := \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(w)$, $\delta' := \epsilon(\pi)(\tau') \uplus \tau'$, and $\bar{w}' := \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta') \uplus \delta')(w')$, we have $\lhd = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta') \uplus \delta')(<')$, $\precapprox = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta') \uplus \delta')(\lesssim')$, $\bar{w}' \lhd^{+} \bar{w}$, $\lhd \circ \precapprox \subseteq \lhd^{+}$, and $\lhd$ is wellfounded.

Since, for all $\Pi \in M$, $\Pi\Delta \in \mathrm{logic}(G)$ is assumed to be $(\delta, e, \mathcal{A})$-valid whereas Δ is assumed to be not, we know that M is $(\delta, e, \mathcal{A})$-valid. By the definition of M, this means that $\Phi\varrho$ is not $(\delta, e, \mathcal{A})$-valid (due to (1)) and (in case of hypothesis application only) $\lhd = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(<'\varrho)$ (due to (4)), $\precapprox = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(\lesssim'\varrho)$ (due to (5)), $\mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(w'\varrho) \lhd \bar{w}$ (due to (2)), $\lhd \circ \precapprox \subseteq \lhd^{+}$ (due to (6)), and $\lhd$ is wellfounded (due to (3)).

Define $\tau'(y^{\forall,\mathrm{w}}) := \left\{ \begin{array}{ll} \epsilon(e)(\delta)(\varrho(y^{\forall,\mathrm{w}})) & \text{for } y^{\forall,\mathrm{w}} \in Y \\ \tau(y^{\forall,\mathrm{w}}) & \text{for } y^{\forall,\mathrm{w}} \in \mathrm{V}_{\forall,\mathrm{w}} \setminus Y \end{array} \right\}$ and $\delta' := \epsilon(\pi)(\tau') \uplus \tau'$.

Claim 2: For $v^{\forall,\mathrm{s}} \in \mathcal{V}_{\forall,\mathrm{s}}(\Phi, \urcorner)$ we have $\epsilon(\pi)(\tau)(v^{\forall,\mathrm{s}}) = \epsilon(\pi)(\tau')(v^{\forall,\mathrm{s}})$.
Proof of Claim 2: Otherwise there must be some $y^{\forall,\mathrm{w}} \in Y$ with $y^{\forall,\mathrm{w}}\ S_\pi\ v^{\forall,\mathrm{s}}$. Since $v^{\forall,\mathrm{s}} \in \mathcal{V}_{\forall,\mathrm{s}}(\Phi, \urcorner)$ we have $v^{\forall,\mathrm{s}}\ R'\ y^{\forall,\mathrm{w}}$ by definition of Y. But then $R' \cup S_e \cup S_\pi$ is not wellfounded, which contradicts π being $(e, \mathcal{A})$-compatible with (C', R'). Q.e.d. (Claim 2)

Claim 3: For $x^{\exists} \in \mathcal{V}_{\exists}(\Phi, \urcorner)$ we have $\epsilon(e)(\delta)(x^{\exists}) = \epsilon(e)(\delta')(x^{\exists})$.
Proof of Claim 3: Otherwise we have $\epsilon(e)(\epsilon(\pi)(\tau) \uplus \tau)(x^{\exists}) \neq \epsilon(e)(\epsilon(\pi)(\tau') \uplus \tau')(x^{\exists})$. Then there must be some $y^{\forall,\mathrm{w}} \in Y$ with $y^{\forall,\mathrm{w}}\ S_\pi \circ S_e\ x^{\exists}$ (i.e. the first occurrence of τ' makes a difference) or $y^{\forall,\mathrm{w}}\ S_e\ x^{\exists}$ when the second occurrence of τ' makes a difference. Since $x^{\exists} \in \mathcal{V}_{\exists}(\Phi, \urcorner)$ we have $x^{\exists}\ R'\ y^{\forall,\mathrm{w}}$ by definition of Y. But then $R' \cup S_e \cup S_\pi$ is not wellfounded, which contradicts π being $(e, \mathcal{A})$-compatible with (C', R'). Q.e.d. (Claim 3)

The resp. values of Φ, w', $<'$, and $\lesssim'$ under $\mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta') \uplus \delta')$ are the same as the values of Φ, w', $<'$, and $\lesssim'$ under $\mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \epsilon(\pi)(\tau) \uplus \tau')$ by definition of δ', Claim 2, and Claim 3, which again are the same as the values of $\Phi\varrho$, $w'\varrho$, $<'\varrho$, and $\lesssim'\varrho$ under $\mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)$ by the Substitution-Lemma and the definition of δ. Thus, due to $\Phi\varrho$ not being $(\delta, e, \mathcal{A})$-valid, $((\Phi, \urcorner), \tau')$ is an $(\pi, e, \mathcal{A})$-counterexample with (in case of hypothesis application only) $\lhd = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(<'\varrho) = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta') \uplus \delta')(<')$, $\precapprox = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(\lesssim'\varrho) = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta') \uplus \delta')(\lesssim')$, and $\bar{w}' = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta)(w'\varrho) \lhd \bar{w}$. **Q.e.d. (Theor. 14.3)**

Proof of Lemma A.1

Here we denote concatenation (product) of relations '$\circ$' simply by juxtaposition and assume it to have higher priority than any other binary operator.

(1): When e is an existential $(\mathcal{A}, R')$-valuation, $R' \cup S_e$ is wellfounded. In case of $R \subseteq R'$, we have $R \cup S_e \subseteq R' \cup S_e$ and $R \cup S_e$ is wellfounded, too.

(2): Set $\sigma' := {}_{\mathrm{V}_\exists \setminus \mathrm{dom}(\sigma)}\upharpoonleft \mathrm{id} \cup {}_{\mathrm{V}_\exists}\upharpoonleft \sigma$. Let e' be an existential $(\mathcal{A}, R')$-valuation. Define $S_e := S_{e'}({}_{\mathrm{V}_\exists \setminus \mathrm{dom}(\sigma)}\upharpoonleft \mathrm{id} \cup E_\sigma\upharpoonright_{\mathrm{V}_\exists}) \cup U_\sigma\upharpoonright_{\mathrm{V}_\exists}$ and the existential $(\mathcal{A}, R)$-valuation e by $(x \in \mathrm{V}_\exists,\ \tau' \in S_e\langle\!\{x\}\!\rangle \to \mathcal{A})$: $e(x)(\tau') := \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\tau) \uplus \tau)(\sigma'(x))$ where $\tau \in \mathrm{V}_\forall \to \mathcal{A}$ is an arbitrary extension of τ'. For this definition to be okay, we have to prove the following claims:

Claim 1: For $x \in \mathrm{V}_\exists$, $y \in \mathcal{V}_\forall(\sigma'(x))$, the choice of $\tau \supseteq \tau'$ does not influence the value of $\tau(y)$.
Claim 2: For $x \in \mathrm{V}_\exists$, $x' \in \mathcal{V}_\exists(\sigma'(x))$, the choice of $\tau \supseteq \tau'$ does not influence the value of $\epsilon(e')(\tau)(x')$.
Claim 3: $R \cup S_e$ is wellfounded.
Proof of Claim 1: $y \in \mathcal{V}_\forall(\sigma'(x))$ means $(y, x) \in U_\sigma\upharpoonright_{\mathrm{V}_\exists}$. By definition of S_e we have $(y, x) \in S_e$, i.e. $y \in S_e\langle\!\{x\}\!\rangle = \mathrm{dom}(\tau')$. Q.e.d. (Claim 1)
Proof of Claim 2: $x' \in \mathcal{V}_\exists(\sigma'(x))$ means $(x', x) \in {}_{\mathrm{V}_\exists \setminus \mathrm{dom}(\sigma)}\upharpoonleft \mathrm{id} \cup E_\sigma\upharpoonright_{\mathrm{V}_\exists}$. Thus by definition of S_e we have $S_{e'}\{(x', x)\} \subseteq S_e$, i.e. $S_{e'}\langle\!\{x'\}\!\rangle \subseteq S_e\langle\!\{x\}\!\rangle = \mathrm{dom}(\tau')$. Therefore $\epsilon(e')(\tau)(x') = e'(x')({}_{S_{e'}\langle\!\{x'\}\!\rangle}\upharpoonleft \tau) = e'(x')({}_{S_{e'}\langle\!\{x'\}\!\rangle}\upharpoonleft \tau')$. Q.e.d. (Claim 2)
Proof of Claim 3: $R' \cup S_{e'}$ is wellfounded because e' is an existential $(\mathcal{A}, R')$-valuation. Moreover, as R' is the σ-update of R, we have[30] $R' = R \cup E_\sigma \cup U_\sigma$. Thus, $(R \cup E_\sigma \cup U_\sigma \cup S_{e'})^+$ is a wellfounded ordering, just like its subset $(R \cup S_{e'}({}_{\mathrm{V}_\exists \setminus \mathrm{dom}(\sigma)}\upharpoonleft \mathrm{id} \cup E_\sigma\upharpoonright_{\mathrm{V}_\exists}) \cup U_\sigma\upharpoonright_{\mathrm{V}_\exists})^+$, which is equal to $(R \cup S_e)^+$. Q.e.d. (Claim 3)
Now, for $\tau \in \mathrm{V}_\forall \to \mathcal{A}$ and $x \in \mathrm{V}_\exists$ we have

$$\epsilon(e)(\tau)(x) = e(x)({}_{S_e\langle\!\{x\}\!\rangle}\upharpoonleft \tau) = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\tau) \uplus \tau)(\sigma'(x)),$$

i.e.

$$\epsilon(e)(\tau) = \sigma' \circ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\tau) \uplus \tau).$$

(3): Set $\sigma' := {}_{\mathrm{V}_\exists \setminus \mathrm{dom}(\sigma)}\upharpoonleft \mathrm{id} \cup {}_{\mathrm{V}_\exists}\upharpoonleft \sigma$.
Define $S_e := (S_{\pi'} \cup {}_{\mathrm{V}_{\forall,\mathrm{w}}}\upharpoonleft \mathrm{id})(S_{e'}({}_{\mathrm{V}_\exists \setminus \mathrm{dom}(\sigma)}\upharpoonleft \mathrm{id} \cup E_\sigma\upharpoonright_{\mathrm{V}_\exists}) \cup U_\sigma\upharpoonright_{\mathrm{V}_\exists})$ and the existential $(\mathcal{A}, R)$-valuation e by $(x \in \mathrm{V}_\exists,\ \tau' \in S_e\langle\!\{x\}\!\rangle \to \mathcal{A})$:

$$e(x)(\tau') := \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\epsilon(\pi')(\tau) \uplus \tau) \uplus \epsilon(\pi')(\tau) \uplus \tau)(\sigma'(x))$$

where $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$ is an arbitrary extension of τ'.
For this definition to be okay, we have to prove the following claims:

Claim 4: For $x \in \mathrm{V}_\exists$ and $y \in \mathcal{V}(\sigma'(x))$, the choice of $\tau \supseteq \tau'$ does not influence:

(a) In case of $y \in \mathrm{V}_{\forall,\mathrm{w}}$, the value of $\tau(y)$.

(b) In case of $y \in \mathrm{V}_{\forall,\mathrm{s}}$, the value of $\epsilon(\pi')(\tau)(y)$.

(c) In case of $y \in \mathrm{V}_\exists$, the value of $\epsilon(e')(\epsilon(\pi')(\tau) \uplus \tau)(y)$.

Claim 5: $R \cup S_e \cup (R' \cup S_{e'} \cup S_{\pi'})^+\upharpoonright_{\mathrm{V}_\forall}$ is wellfounded.

Proof of Claim 4: Exercise! Q.e.d. (Claim 4)

Proof of Claim 5: $R' \cup S_{e'} \cup S_{\pi'}$ is wellfounded because π' is $(e', \mathcal{A})$-compatible with (C', R'). Moreover, as R' is the σ-update of R, we have[31] $R' = R \cup E_\sigma \cup U_\sigma$. Thus, $(R \cup E_\sigma \cup U_\sigma \cup R' \cup S_{e'} \cup S_{\pi'})^+$ is a wellfounded ordering, just like its subset

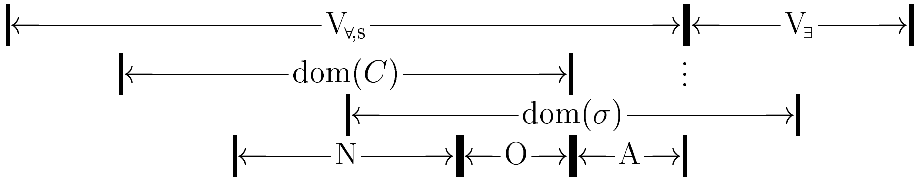
$$(R \cup (S_{\pi'} \cup {}_{\mathrm{V}_{\forall,\mathrm{w}}}\upharpoonleft \mathrm{id})(S_{e'}({}_{\mathrm{V}_\exists \setminus \mathrm{dom}(\sigma)}\upharpoonleft \mathrm{id} \cup E_\sigma\upharpoonright_{\mathrm{V}_\exists}) \cup U_\sigma\upharpoonright_{\mathrm{V}_\exists}) \cup (R' \cup S_{e'} \cup S_{\pi'})^+\upharpoonright_{\mathrm{V}_\forall})^+,$$

which is equal to $(R \cup S_e \cup (R' \cup S_{e'} \cup S_{\pi'})^{+}{\upharpoonleft}_{\mathrm{V}_\forall})^{+}$. Q.e.d. (Claim 5)

Q.e.d. (Lemma A.1)

Proof of Lemma A.2

Assuming σ, C, R, C', R', O, N, $\mathcal{A}$, e', π' as described in the lemma, we set $A := (\mathrm{V}_{\forall,\mathrm{s}} \cap \mathrm{dom}(\sigma)) \setminus (N \uplus O)$. As σ is quasi-existential, we have $\mathrm{V}_\forall \cap \mathrm{dom}(\sigma) \subseteq N \uplus O \uplus A \subseteq \mathrm{V}_{\forall,\mathrm{s}}$. This leaves us in the following situation:



Note that we are not really interested in A because the variables in A should not occur anywhere and thus their substitution is not of any interest. But, as we do not add requirements on this, we have to include the possibility of a nonempty A, although we do not spend much thought on A in this proof.

Note that C' is an R'-choice-condition due to Lemma 9.6.

As π' is $(e', \mathcal{A})$-compatible with (C', R'),

$$\lhd := (R' \cup S_{e'} \cup S_{\pi'})^{+}$$

is a wellfounded ordering.

Let e be the existential $(\mathcal{A}, R)$-valuation given by Lemma A.1(3) for e'. Then

$$S_e \quad = \quad (S_{\pi'} \cup {}_{\mathrm{V}_{\forall,\mathrm{w}}}\rceil\mathrm{id}) \ \circ \ (S_{e'} \circ ({}_{\mathrm{V}_\exists \setminus \mathrm{dom}(\sigma)}\rceil\mathrm{id} \cup E_\sigma{\upharpoonleft}_{\mathrm{V}_\exists}) \cup U_\sigma{\upharpoonleft}_{\mathrm{V}_\exists}) \tag{A.2.1}$$

and for all $\delta \in \mathrm{V}_\forall \to \mathcal{A}$, when τ abbreviates ${}_{\mathrm{V}_{\forall,\mathrm{w}}}\rceil\delta$:

$$\epsilon(e)(\delta) \quad = \quad ({}_{\mathrm{V}_\exists \setminus \mathrm{dom}(\sigma)}\rceil\mathrm{id} \cup {}_{\mathrm{V}_\exists}\rceil\sigma) \ \circ \ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\epsilon(\pi')(\tau) \uplus \tau) \uplus \epsilon(\pi')(\tau) \uplus \tau) \tag{A.2.2}$$

and

$$R \cup S_e \cup \lhd{\upharpoonleft}_{\mathrm{V}_\forall} \ \text{ is wellfounded.} \tag{A.2.3}$$

<u>Claim 1:</u> For any term or formula B (possibly containing some unbound variables from a set $W \subseteq \mathrm{V}_{\mathrm{bound}}$) and any $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$, $\chi \in W \to \mathcal{A}$, and $\delta, \delta', \bar{\bar{\delta}}' \in \mathrm{V}_\forall \to \mathcal{A}$ with ${}_{\mathrm{V}_{\forall,\mathrm{w}}}\rceil\delta = \tau$, ${}_{\mathcal{V}_\forall(\langle\mathcal{V}_\exists(B)\rangle\sigma)}\rceil\bar{\bar{\delta}}' = {}_{\mathcal{V}_\forall(\langle\mathcal{V}_\exists(B)\rangle\sigma)}\rceil\delta'$, $\delta' = \epsilon(\pi')(\tau) \uplus \tau$:

$$\begin{aligned} & \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \bar{\bar{\delta}}' \uplus \chi)(B\sigma) \\ = \ & \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus {}_{\mathrm{V}_\forall}\rceil\sigma \circ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \bar{\bar{\delta}}') \uplus {}_{\mathrm{V}\setminus\mathrm{dom}(\sigma)}\rceil\bar{\bar{\delta}}' \uplus \chi)(B). \end{aligned}$$

<u>Proof of Claim 1:</u> $\mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \bar{\bar{\delta}}' \uplus \chi)(B\sigma) =$ (by the Substitution-Lemma)
$\mathrm{eval}(\mathcal{A} \uplus ({}_{\mathrm{V}\setminus\mathrm{dom}(\sigma)}\rceil\mathrm{id} \uplus \sigma) \circ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \bar{\bar{\delta}}' \uplus \chi))(B) =$ (by the Explicitness-Lemma (as the variables of W do not occur free in $\mathrm{ran}(\sigma)$) and by ${}_{\mathcal{V}_\forall(\langle\mathcal{V}_\exists(B)\rangle\sigma)}\rceil\bar{\bar{\delta}}' = {}_{\mathcal{V}_\forall(\langle\mathcal{V}_\exists(B)\rangle\sigma)}\rceil\delta'$)

$$\mathrm{eval}\left(\begin{array}{ll} & \mathcal{A} \\ \uplus & ({}_{\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma)}\rceil\mathrm{id} \uplus {}_{\mathrm{V}_\exists}\rceil\sigma) \circ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \delta') \\ \uplus & {}_{\mathrm{V}_\forall}\rceil\sigma \circ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \bar{\bar{\delta}}') \\ \uplus & {}_{\mathrm{V}\setminus\mathrm{dom}(\sigma)}\rceil\bar{\bar{\delta}}' \uplus \chi \end{array}\right)(\ B\ ) =$$

(by (A.2.2), ${}_{\mathrm{V}_{\forall,\mathrm{w}}}\rceil\delta = \tau$, and $\delta' = \epsilon(\pi')(\tau) \uplus \tau$)
$\mathrm{eval}(\epsilon(e)(\delta) \uplus {}_{\mathrm{V}_\forall}\rceil\sigma \circ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \bar{\bar{\delta}}') \uplus {}_{\mathrm{V}\setminus\mathrm{dom}(\sigma)}\rceil\bar{\bar{\delta}}' \uplus \chi)(B)$. <u>Q.e.d. (Claim 1)</u>

Claim 2: ${}_{\mathrm{V}_\forall}{\upharpoonleft}(R^{+}) \subseteq \lhd$, $\quad U_\sigma{\circ}R^{+} \subseteq \lhd$, and $S_e{\circ}R^{+} \subseteq \lhd$.
Proof of Claim 2: As R' is the σ-update of R, we have[32] $R' = R \cup E_\sigma \cup U_\sigma$. Thus, Claim 2 follows from (A.2.1). Q.e.d. (Claim 2)

Set $S_\pi := \lhd \cap (\mathrm{V}_{\forall,\mathrm{w}} \times \mathrm{V}_{\forall,\mathrm{s}})$.

Claim 3: $R \cup S_e \cup S_\pi$ is wellfounded.
Proof of Claim 3: This follows from (A.2.3). Q.e.d. (Claim 3)

The idea for the definition of the π we have to find is—roughly speaking—as follows: For $y^{\forall,\mathrm{s}} \notin N{\uplus}O{\uplus}A$ we take $\pi(y^{\forall,\mathrm{s}})$ to be $\pi'(y^{\forall,\mathrm{s}})$. For $y^{\forall,\mathrm{s}} \in O$ we evaluate $\sigma(y^{\forall,\mathrm{s}})$ in the updated environment because we know that $\langle O\rangle Q_{C,\sigma}$ is valid there by assumption of the lemma. For $y^{\forall,\mathrm{s}} \in A$ we take the same because we do not think much about this case. For $y^{\forall,\mathrm{s}} \in N$, however, we have to take care of $(e, \mathcal{A})$-compatibility with (C, R) explicitly in an $\lhd$-recursive definition.

Let π be defined by $(y^{\forall,\mathrm{s}} \in \mathrm{V}_{\forall,\mathrm{s}},\ \ \tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A})$

$$\pi(y^{\forall,\mathrm{s}})({}_{S_\pi\langle\{y^{\forall,\mathrm{s}}\}\rangle}{\upharpoonleft}\tau) := \begin{cases} f & \text{if } y^{\forall,\mathrm{s}} \in N \\ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\epsilon(\pi')(\tau) \uplus \tau) \uplus \epsilon(\pi')(\tau) \uplus \tau)(\sigma(y^{\forall,\mathrm{s}})) & \text{if } y^{\forall,\mathrm{s}} \in O{\uplus}A \\ \pi'(y^{\forall,\mathrm{s}})({}_{S_{\pi'}\langle\{y^{\forall,\mathrm{s}}\}\rangle}{\upharpoonleft}\tau) & \text{otherwise} \end{cases}$$

where (for details cf. the proof of Lemma 9.3) f is chosen s.t., for $C(y^{\forall,\mathrm{s}}) = \lambda v_0.\ \ldots \lambda v_{l-1}.\ B$, and for any $\chi \in \{v_0, \ldots, v_{l-1}\} \to \mathcal{A}$

B becomes—if possible— $({}_{\mathrm{V}_{\forall,\mathrm{s}}\setminus\{y^{\forall,\mathrm{s}}\}}{\upharpoonleft}(\epsilon(\pi)(\tau)) \uplus \{y^{\forall,\mathrm{s}}{\mapsto}f\} \uplus \tau \uplus \chi, e, \mathcal{A})$-valid.

Note that this definition is okay because the only part of τ that is relevant on the right-hand side is ${}_{S_\pi\langle\{y^{\forall,\mathrm{s}}\}\rangle}{\upharpoonleft}\tau$ and because it is recursive in $\lhd$; indeed, we have $(E_\sigma \cup U_\sigma){\restriction}_{\mathrm{V}_\forall} \subseteq R'$ due to R' being the σ-update of R, and for $x^{\exists} \in \mathcal{V}_{\exists}(C(y^{\forall,\mathrm{s}}))$ we have $x^{\exists}\ R^{+}\ y^{\forall,\mathrm{s}}$ (as C is an R-choice-condition) and then for $v^{\forall}\ S_e\ x^{\exists}$ we have $v^{\forall} \lhd y^{\forall,\mathrm{s}}$ by Claim 2, and for $z^{\forall} \in \mathcal{V}_{\forall}(C(y^{\forall,\mathrm{s}}))\setminus\{y^{\forall,\mathrm{s}}\}$ we have $z^{\forall}\ R^{+}\ y^{\forall,\mathrm{s}}$ $\ z^{\forall} \lhd y^{\forall,\mathrm{s}}$ by Claim 2.

Claim 4: For all $y^{\forall,\mathrm{s}} \in O{\uplus}A$ and $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$, when we set $\delta' := \epsilon(\pi')(\tau) \uplus \tau$:

$$\epsilon(\pi)(\tau)(y^{\forall,\mathrm{s}}) = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \delta')(\sigma(y^{\forall,\mathrm{s}})).$$

Proof of Claim 4: Immediately by the definition of π. Q.e.d. (Claim 4)

Claim 5: For all $y^{\forall,\mathrm{s}} \in \mathrm{V}_{\forall,\mathrm{s}}\setminus(N{\uplus}O{\uplus}A)$ and $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$: $\ \epsilon(\pi)(\tau)(y^{\forall,\mathrm{s}}) = \epsilon(\pi')(\tau)(y^{\forall,\mathrm{s}})$.
Proof of Claim 5: Immediately by the definition of π. Q.e.d. (Claim 5)

Claim 6: For any term or formula B (possibly containing some unbound variables from a set $W \subseteq \mathrm{V}_{\mathrm{bound}}$) with $N{\cap}\mathcal{V}(B) = \emptyset$, and for any $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$ and $\chi \in W \to \mathcal{A}$, when we set $\delta := \epsilon(\pi)(\tau) \uplus \tau$ and $\delta' := \epsilon(\pi')(\tau) \uplus \tau$, we have

$$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \delta' \uplus \chi)(B\sigma) = \mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta \uplus \chi)(B).$$

Proof of Claim 6: $\mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \delta' \uplus \chi)(B\sigma) =$ (by Claim 1)
$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus {}_{\mathrm{V}_\forall}{\upharpoonleft}\sigma \circ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \delta') \ \uplus {}_{\mathrm{V}\setminus\mathrm{dom}(\sigma)}{\upharpoonleft}\delta' \uplus \chi)(B) =$
(by $O{\uplus}A \subseteq \mathrm{V}_\forall \cap \mathrm{dom}(\sigma) \subseteq N{\uplus}O{\uplus}A$ and $N{\cap}\mathcal{V}(B) = \emptyset$)
$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus {}_{O\uplus A}{\upharpoonleft}\sigma \circ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \delta') \uplus {}_{\mathrm{V}\setminus(N\uplus O\uplus A)}{\upharpoonleft}\delta' \ \uplus \chi)(B) =$
(by Claim 4 and Claim 5)
$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta \uplus \chi)(B)$. Q.e.d. (Claim 6)

Claim 7: For any set of sequents G' (possibly containing some unbound variables from a set $W \subseteq \mathrm{V}_{\mathrm{bound}}$) with $N{\cap}\mathcal{V}(G') = \emptyset$, and for any $\tau \in (\mathrm{V}_{\forall,\mathrm{w}} \cup W) \to \mathcal{A}$:
$(\epsilon(\pi)(\tau){\uplus}\tau, e, \mathcal{A})$-validity of G' is logically equivalent to $(\epsilon(\pi')(\tau){\uplus}\tau, e', \mathcal{A})$-validity of $G'\sigma$.
Proof of Claim 7: This is a trivial consequence of Claim 6. Q.e.d. (Claim 7)

Claim 8: For $y^{\forall,\mathrm{s}} \in \mathrm{dom}(C) \setminus N$ we have $N \cap \mathcal{V}(C(y^{\forall,\mathrm{s}})) = \emptyset$.
Proof of Claim 8: Otherwise there is some $z^{\forall,\mathrm{s}} \in N \cap \mathcal{V}(C(y^{\forall,\mathrm{s}}))$, but then $z^{\forall,\mathrm{s}}\ R^*\ y^{\forall,\mathrm{s}}$ as C is an R-choice-condition, and then, as $\mathrm{dom}(C) \cap \langle N \rangle R^+ \subseteq N$, we would have the contradicting $y^{\forall,\mathrm{s}} \in N$. Q.e.d. (Claim 8)

Claim 9: Let $y^{\forall,\mathrm{s}} \in \mathrm{dom}(C)$ and $C(y^{\forall,\mathrm{s}}) = \lambda v_0.\ \ldots \lambda v_{l-1}.\ B$. Let $\tau \in \mathrm{V}_{\forall,\mathrm{w}} \to \mathcal{A}$ and $\chi \in \{v_0, \ldots, v_{l-1}\} \to \mathcal{A}$ and suppose that, for some $\eta \in \{y^{\forall,\mathrm{s}}\} \to \mathcal{A}$, B is $(\bar{\delta}, e, \mathcal{A})$-valid for $\bar{\delta} := {}_{\mathrm{V}_{\forall,\mathrm{s}} \setminus \{y^{\forall,\mathrm{s}}\}}{\upharpoonleft}(\epsilon(\pi)(\tau)) \uplus \eta \uplus \tau \uplus \chi$. Now: B is $(\delta, e, \mathcal{A})$-valid for $\delta := \epsilon(\pi)(\tau) \uplus \tau \uplus \chi$.

Proof of Claim 9: Set $\bar{\delta}' := {}_{\mathrm{V}_{\forall,\mathrm{s}} \setminus \{y^{\forall,\mathrm{s}}\}}{\upharpoonleft}(\epsilon(\pi')(\tau)) \uplus \eta \uplus \tau \uplus \chi$ and $\delta' := \epsilon(\pi')(\tau) \uplus \tau \uplus \chi$.

$y^{\forall,\mathrm{s}} \notin O \uplus N$: In this case, we have $y^{\forall,\mathrm{s}} \notin \mathrm{dom}(\sigma)$ because of $\mathrm{dom}(C) \cap \mathrm{dom}(\sigma) \subseteq O \uplus N$. Thus, as (C', R') is the extended σ-update of (C, R), we have $C'(y^{\forall,\mathrm{s}}) = (C(y^{\forall,\mathrm{s}}))\sigma$. By Claim 8 we have $N \cap \mathcal{V}(B) = \emptyset$. For later application of Claim 1, note that ${}_{\mathcal{V}_\forall(\langle \mathcal{V}_\exists(B) \rangle \sigma)}{\upharpoonleft}\bar{\delta}' = {}_{\mathcal{V}_\forall(\langle \mathcal{V}_\exists(B) \rangle \sigma)}{\upharpoonleft}\delta'$; otherwise there would be some $x^\exists \in \mathcal{V}_\exists(B) = \mathcal{V}_\exists(C(y^{\forall,\mathrm{s}}))$ with $y^{\forall,\mathrm{s}}\ U_\sigma\ x^\exists$, and then, as C is a R-choice-condition, $y^{\forall,\mathrm{s}}\ U_\sigma \circ R^+\ y^{\forall,\mathrm{s}}$, and then, by Claim 2, $y^{\forall,\mathrm{s}} \lhd y^{\forall,\mathrm{s}}$, which contradicts the wellfoundedness of $\lhd$.
Note that ${}_{\mathcal{V}_{\forall,\mathrm{s}}(B)}{\upharpoonleft}\sigma \circ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \delta') = {}_{\mathcal{V}_{\forall,\mathrm{s}}(B)}{\upharpoonleft}\sigma \circ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \bar{\delta}')$; otherwise there would be some $z^{\forall,\mathrm{s}} \in \mathcal{V}_{\forall,\mathrm{s}}(C(y^{\forall,\mathrm{s}}))$ with $y^{\forall,\mathrm{s}} \in \mathcal{V}(\sigma(z^{\forall,\mathrm{s}}))$, which implies $y^{\forall,\mathrm{s}}\ R'\ z^{\forall,\mathrm{s}}\ R^*\ y^{\forall,\mathrm{s}}$ (as R' is the σ-update of R and C is an R-choice-condition), and then, by Claim 2, $y^{\forall,\mathrm{s}} \lhd z^{\forall,\mathrm{s}} \unlhd y^{\forall,\mathrm{s}}$, which contradicts the wellfoundedness of $\lhd$.
${}_{\mathcal{V}(B)}{\upharpoonleft}\bar{\delta} =$ (due to $y^{\forall,\mathrm{s}} \notin \mathrm{dom}(\sigma)$, $N \cup (\mathrm{dom}(\sigma) \cap \mathrm{V}_\forall) = N \uplus O \uplus A$, $N \cap \mathcal{V}(B) = \emptyset$, Claim 5)
${}_{\mathcal{V}(B) \setminus \mathrm{dom}(\sigma)}{\upharpoonleft}\bar{\delta}' \ \uplus\ {}_{(O \uplus A) \cap \mathcal{V}_{\forall,\mathrm{s}}(B)}{\upharpoonleft}(\epsilon(\pi)(\tau)) =$ (by Claim 4)
${}_{\mathcal{V}(B) \setminus \mathrm{dom}(\sigma)}{\upharpoonleft}\bar{\delta}' \ \uplus\ {}_{(O \uplus A) \cap \mathcal{V}_{\forall,\mathrm{s}}(B)}{\upharpoonleft}\sigma \circ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \delta') =$ (cf. above)
${}_{\mathcal{V}(B) \setminus \mathrm{dom}(\sigma)}{\upharpoonleft}\bar{\delta}' \ \uplus\ {}_{(O \uplus A) \cap \mathcal{V}_{\forall,\mathrm{s}}(B)}{\upharpoonleft}\sigma \circ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \bar{\delta}')$.
Now: $\mathsf{TRUE} =$ (by assumption of Claim 9)
$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\bar{\delta}) \uplus \bar{\delta})(B) =$ (by the above and $\mathrm{dom}(\sigma) \cap \mathcal{V}_\forall(B) = (O \uplus A) \cap \mathcal{V}_{\forall,\mathrm{s}}(B)$)
$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e)(\bar{\delta}) \uplus {}_{\mathrm{V}_\forall}{\upharpoonleft}\sigma \circ \mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \bar{\delta}') \uplus {}_{\mathrm{V} \setminus \mathrm{dom}(\sigma)}{\upharpoonleft}\bar{\delta}')(B) =$ (by Claim 1)
$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\delta') \uplus \bar{\delta}')(B\sigma) =$ (as otherwise for some $x^\exists \in \mathcal{V}_\exists(B\sigma) = \mathcal{V}_\exists(C'(y^{\forall,\mathrm{s}}))$ we have $y^{\forall,\mathrm{s}}\ S_{e'}\ x^\exists\ R'^+\ y^{\forall,\mathrm{s}}$, i.e. $y^{\forall,\mathrm{s}} \lhd y^{\forall,\mathrm{s}}$)
$\mathrm{eval}(\mathcal{A} \uplus \epsilon(e')(\bar{\delta}') \uplus \bar{\delta}')(B\sigma)$. As π' is $(e', \mathcal{A})$-compatible with (C', R'), we know that $B\sigma$ is $(\delta', e', \mathcal{A})$-valid. Thus, by Claim 7, B is $(\delta, e, \mathcal{A})$-valid.

$y^{\forall,\mathrm{s}} \in O$: $N \cap \mathcal{V}(B) = \emptyset$ by Claim 8. Let $y \in \mathrm{V}_{\mathrm{bound}} \setminus \mathcal{V}(C(y^{\forall,\mathrm{s}}))$ and D be the formula $\exists y.\ (B\{y^{\forall,\mathrm{s}}(v_0) \cdots (v_{l-1}) \mapsto y\})$ s.t. $Q_{C,\sigma}(y^{\forall,\mathrm{s}})$ is equal to $\forall v_0.\ \ldots \forall v_{l-1}.\ (D \Rightarrow B)\sigma$. We have $N \cap \mathcal{V}(D) = \emptyset$. As B is valid in $\mathcal{A} \uplus \epsilon(e)(\bar{\delta}) \uplus \bar{\delta}$, by the standard semantical definition of $\exists$ (cf. e.g. Wirth (1997), p. 188, or Enderton (1973), p. 82) and the Substitution-Lemma, D is valid in $\mathcal{A} \uplus \epsilon(e)(\bar{\delta}) \uplus \bar{\delta}$, too, and then (as $y^{\forall,\mathrm{s}}$ does not occur in D anymore (as all occurrences of $y^{\forall,\mathrm{s}}$ in B are of the form $y^{\forall,\mathrm{s}}(v_0) \cdots (v_{l-1})$ according to Definition 9.1)) also valid in $\mathcal{A} \uplus \epsilon(e)(\bar{\delta}) \uplus \delta$. Moreover, D is even valid in $\mathcal{A} \uplus \epsilon(e)(\delta) \uplus \delta$; otherwise there would be some $v^\exists \in \mathcal{V}(D)$ with $y^{\forall,\mathrm{s}}\ S_e\ v^\exists$, but then $v^\exists \in \mathcal{V}(C(y^{\forall,\mathrm{s}})) \setminus \{y^{\forall,\mathrm{s}}\}$ and (as C is an R-choice-condition) $v^\exists\ R^+\ y^{\forall,\mathrm{s}}$, which contradicts the wellfoundedness of $R \cup S_e$, which contradicts e being an existential $(\mathcal{A}, R)$-valuation. By Claim 7, $D\sigma$ is $(\delta', e', \mathcal{A})$-valid. But by assumption of the lemma on $Q_{C,\sigma}$ and by the standard definition of $\forall$, we know that $(D \Rightarrow B)\sigma$ is $(\delta', e', \mathcal{A})$-valid. Thus, $B\sigma$ is $(\delta', e', \mathcal{A})$-valid. By Claim 7, B is $(\delta, e, \mathcal{A})$-valid.

$y^{\forall,\mathrm{s}} \in N$: By definition of π. Q.e.d. (Claim 9)

By Claim 3 and Claim 9, π is $(e, \mathcal{A})$-compatible with (C, R), and then items 1 and 2 of the lemma are trivial consequences of Claim 6, Claim 7, resp.. **Q.e.d. (Lemma A.2)**

C Notes

Note 1: Indeed, despite Sieg & Byrnes (1998), the old fashion of Shanin &al. (1965), namely to find a proof in a sequent calculus in disguise and then to translate it into natural deduction, still seems to be the state of the art of proof search in natural deduction in classical logic.

Note 2: We attribute α, β, γ, and δ to *inference rules* (as it seems to be intended in Smullyan (1968)) and not to formulas (as in Fitting (1996)). This has the advantage that nothing changes when we switch from positive to refutational theorem proving, e.g. β always stands for "branching", as already indicated in Smullyan (1968), contrary to the strange branching α-rules in Fitting (1996), p. 51.

Note 3: Note that $x^{\exists} \notin \mathcal{V}_{\exists}(\mathcal{F})$ is not required for soundness or safeness (cf. the proof of Theor. 14.1), but only in order not to lose possible proofs.

Note 4: Note that for soundness and safeness of the δ-rule $x^{\forall,\mathrm{w}} \notin \mathcal{V}(A, \Gamma\Pi, \sqsupset) \cup \mathrm{dom}(R)$ is sufficient for $\mathcal{F} = (F, C, R, L, H)$, cf. the proof of Theor. 14.1. Nevertheless, we require $x^{\forall,\mathrm{w}} \notin \mathcal{V}(\mathcal{F})$ in order not to lose possible proofs.

Note 5: Note that for soundness and safeness of the liberalized δ-rule $x^{\forall,\mathrm{s}} \notin \mathcal{V}(A) \cup \mathrm{dom}(C{\cup}R)$ is sufficient for $\mathcal{F} = (F, C, R, L, H)$, cf. the proof of Theor. 14.1. Nevertheless, we require $x^{\forall,\mathrm{s}} \notin \mathcal{V}(\mathcal{F})$ in order not to lose possible proofs.

Note 6: The notation $\langle A \rangle R$ is in the tradition of Bourbaki (1954), Chapitre II, § 3, Définition 3, where $R\langle A \rangle$ is written in order to clearly distinguish relation application $\langle A \rangle R$ from function application $R(A)$. In Wirth (1997) we still used to write $R[A]$ instead of $\langle A \rangle R$. In this paper, however, this notation would lead to confusion with our use of optional brackets.

Note 7: Note that R^{-1} is an *inverse* (in the sense that $R{\circ}R^{-1} = {}_{\mathrm{dom}(R)}{\upharpoonleft}\mathrm{id}$ and $R^{-1}{\circ}R = {}_{\mathrm{ran}(R)}{\upharpoonleft}\mathrm{id}$ holds) iff R is an injective function.

Note 8: We do not need the more complicated definitions of a sequent as a pair of lists of formulas or as a T/F-tagged list of formulas because we do not consider calculi where the separation of a sequent into antecedent and succedent is important, like LJ in Gentzen (1935) or the "symmetric Gentzen systems" in Smullyan (1968).

Note 9: The name "syntactical construct" was already used before Wirth & Becker (1995) and we will keep it in order to avoid extra confusion until we find a really better one.

Note 10: It may be objected that in the modal logics of, say, Fitting (1999), Cerrito & Cialdea (2001), Fitting (2002), the Substitution-Lemma is not valid because it only holds for the substitution of rigid and rigidified (grounded, annotated, non-relativized) terms. This is, however, a wrong view: Those substitutions for that the Substitution-Lemma does not hold are no proper substitutions. They cannot occur in proof steps because such proof steps would be unsound. And therefore we do not need them at all, and simply do not call them substitutions, which renders the Substitution-Lemma valid again. Indeed, the substitutions for that the Substitution-Lemma does not hold when applied to a certain term or formula B, are not "free" for B in some sense. The problem is that an implicit variable is captured by some quantifier. We will explain this for the higher-order modal logic of Fitting (2002) because there the relativization operator $\downarrow$ makes this obvious. For a term t of intensional type $\uparrow\alpha$, the term $\downarrow t$ has the extensional type α. Instead of $\downarrow t$ one could also write tw where w is a variable valuated to the current world, so that tw is the extension of t at world w. The quantifiers $\Box$, $\Diamond$ and the binder λ implicitly bind this implicit variable w. Let us now have a look on the standard example for the violation of the Substitution-Lemma. Let x, y be variables of the extensional type 0, let h, p be constants of the intensional type $\uparrow 0$ standing for the intentional notions of Hesperus (morning star) and phosphorus (evening star), and assume that $\Box$ means "the ancients knew". Then

$$x = y \quad\Rightarrow\quad \Box(x = y)$$

is valid because the ancients knew that two identical things are identical. On the other hand its instance

$$\downarrow h = \downarrow p \quad\Rightarrow\quad \Box(\downarrow h = \downarrow p)$$

via the "substitution" $\{x \mapsto \downarrow h,\quad y \mapsto \downarrow p\}$ is not valid in our world because here the extensions of Hesperus and phosphorus are identical but the ancients did not know that. But with the variable w made explicit, the first formula reads

$$x = y \quad\Rightarrow\quad \Box w.\ (x = y)$$

for which the "substitution" $\{x \mapsto hw,\quad y \mapsto pw\}$ is obviously not "free" because the w is captured by the quantifier $\Box w$.

Note 11: Indeed, the problems of higher-order logic do not interfere because we do not Skolemize, the word "completeness" will not occur anymore in the rest of the whole paper, and we consider efficiency (like unification) only in so far as that we do not inhibit it.

Note 12: $x^{\forall,\mathrm{w}} \in \mathrm{V}_{\forall,\mathrm{w}}$ is in *solved form* in the syntactical construct $\varGamma\ (x^{\forall,\mathrm{w}} \neq t)\ \varPi;\ \sqsupset$ if $x^{\forall,\mathrm{w}} \notin \mathcal{V}(t, \varGamma\varPi, \sqsupset)$ and $\mathcal{V}_{\forall,\mathrm{s}}(t, \varGamma\varPi, \sqsupset) \cup \mathcal{V}_{\exists}(t, \varGamma\varPi, \sqsupset) \subseteq R^{+}\langle\{x^{\forall,\mathrm{w}}\}\rangle$.

Note 13: In order to show that there is nothing out of the ordinary with lexicographic combination up to m, let us indicate how it could be modeled with fixed-arity many-sorted functions in the specification formalism of QuodLibet.

For $m = 2$ the lexicographic combination up to m can be inductively defined in the following way that can be easily generalized to any natural number m. In our signature we need, for any user-defined type s, the constructor symbol $\mathsf{lex} : \mathsf{nat} \to s \to s \to \mathsf{ORD}$ representing the lexicographic combination of length 0, 1, or 2 as indicated by the first argument, e.g. $\mathsf{lex}(\mathsf{s}(\mathsf{0}), x, y)$ models the 1-tuple (x) while $\mathsf{lex}(\mathsf{0}, x, y)$ models the 0-tuple or empty word (). Moreover, $\mathsf{lex}(\mathsf{s}(\mathsf{s}(n)), x, y)$ models the $n{+}2$-tuple xy^{n+1}, where only $n = 0$ is intended, but $n \succ 0$ does not destroy wellfoundedness. It is important that s is different from ORD because otherwise we can form tuples of arbitrary length via the old idea of set theory to construct tuples from pairs, namely $(x_{n+2}, x_{n+1}, \ldots, x_0) := (x_{n+2}, (x_{n+1}, \ldots, x_0))$, thereby destroying the wellfoundedness. Now we can inductively define $< : \mathsf{ORD} \to \mathsf{ORD} \to \mathsf{bool}$ as follows, where $n_i : \mathsf{nat}$ and $x_i, y_i : s$.

($\lesssim$1) $\quad\forall n_1, x_0, x_1, y_0, y_1.\quad \mathsf{lex}(\mathsf{0}, x_0, y_0) \lesssim \mathsf{lex}(n_1, x_1, y_1)$

($\lesssim$2) $\quad\forall n_0, x_0, x_1, y_0, y_1.\quad \neg(\ \mathsf{lex}(\mathsf{s}(n_0), x_0, y_0) \lesssim \mathsf{lex}(\mathsf{0}, x_1, y_1)\ )$

$$(\lesssim 3)\quad \forall n_0,n_1,x_0,x_1,y_0,y_1.\ \left(\begin{array}{ll} & \mathsf{lex}(\mathsf{s}(n_0),x_0,y_0)\lesssim\mathsf{lex}(\mathsf{s}(n_1),x_1,y_1)\\ \Leftrightarrow & \left(\begin{array}{ll} & x_0\lesssim x_1\\ \wedge & (\ x_1\lesssim x_0 \Rightarrow\ \mathsf{lex}(n_0,y_0,y_0)\lesssim\mathsf{lex}(n_1,y_1,y_1)\ )\end{array}\right)\end{array}\right)$$

$$(<1)\quad \forall n_0,n_1,x_0,x_1,y_0,y_1.\ \left(\begin{array}{ll} & \mathsf{lex}(n_0,x_0,y_0)<\mathsf{lex}(n_1,x_1,y_1)\\ \Leftrightarrow & \left(\begin{array}{ll} & \mathsf{lex}(n_0,x_0,y_0)\lesssim\mathsf{lex}(n_1,x_1,y_1)\\ \wedge & \neg(\ \mathsf{lex}(n_1,x_1,y_1)\lesssim\mathsf{lex}(n_0,x_0,y_0)\ )\end{array}\right)\end{array}\right)$$

($\lesssim$1) and ($\lesssim$2) say that the empty tuple is smallest. ($\lesssim$3) says that for non-empty tuples we compare the first elements and have to compare the rest of the tuples when they turn out to be equivalent. ($<$1) is just the standard way to define the ordering of a quasi-ordering. The analogous definitions for boolean functions are admissible in QuodLibet. Note that we get the following theorems that have been proved in QuodLibet deductively and without using any ordering properties:

$$(<2)\qquad \forall n_1,x_0,x_1,y_0,y_1.\ \ \mathsf{lex}(0,x_0,y_0)<\mathsf{lex}(\mathsf{s}(n_1),x_1,y_1)$$

$$(<3)\quad \forall n_0,n_1,x_0,x_1,y_0,y_1.\ \left(\begin{array}{ll} & \mathsf{lex}(\mathsf{s}(n_0),x_0,y_0)<\mathsf{lex}(\mathsf{s}(n_1),x_1,y_1)\\ \Leftarrow & \left(\begin{array}{ll} & x_0\lesssim x_1\\ \wedge & \left(\begin{array}{ll} & x_1\lesssim x_2\\ \Rightarrow & \mathsf{lex}(n_0,y_0,y_0)<\mathsf{lex}(n_1,y_1,y_1)\end{array}\right)\end{array}\right)\end{array}\right)$$

Using reflexivity of $\lesssim$ on the user-defined types and the definition of $<$ as the ordering of $\lesssim$, we get the following consequences of ($<$3) that are implemented (besides ($<$2)) as special inference rules in QuodLibet instead of ($<$3) since the quasi-orderings are not implemented in QuodLibet for pragmatic reasons. The orderings, however, are available for any type s of the signature Σ via predicate symbols $<: s\to s\to \mathsf{bool}$ of Σ.

$$(<4)\quad \forall n_0,n_1,x_0,x_1,y_0,y_1.\ \left(\begin{array}{ll} & \mathsf{lex}(\mathsf{s}(n_0),x_0,y_0)<\mathsf{lex}(\mathsf{s}(n_1),x_1,y_1)\\ \Leftarrow & x_0<x_1\end{array}\right)$$

$$(<5)\quad \forall n_0,n_1,x_0,y_0,y_1.\ \left(\begin{array}{ll} & \mathsf{lex}(\mathsf{s}(n_0),x_0,y_0)<\mathsf{lex}(\mathsf{s}(n_1),x_0,y_1)\\ \Leftarrow & \mathsf{lex}(n_0,y_0,y_0)<\mathsf{lex}(n_1,y_1,y_1)\end{array}\right)$$

Note 14: An anonymous referee of a previous version of this text wrote:

> A minor item: After stating the relevant induction principle the author writes: "Now by the Principle of Dependent Choice (cf. Rubin & Rubin (1985)) ...." I find this reference quite inappropriate: Of course, one needs some form of the axiom of choice to prove the existence of minimal elements *in general*, however in the context of inductive reasoning the used ordering is always *concretely given* and consequently the fact that "a class without minimal elements contains a chain without a least element" is always obvious in any particular scenario of theorem proving.

I do not follow this argumentation:

> The problem is that there may be several counterexamples and the induction ordering be partial. So you have to pick and pick and pick smaller counterexamples from unstructured non-empty classes.

Nevertheless, it was the above remark which finally made me change my definition of well-foundedness from non-termination of the reverse relation to existence of minimal elements, which resulted in a tautological soundness of the Method of Descente Infinie.

Note 15: Consider the valid Henkin quantified IF logic formula

$$\forall x_0.\ \forall x_1.\ \exists y_0/x_1.\ \exists y_1/x_0.\ (\ x_0{=}y_0 \wedge x_1{=}y_1\ )$$

or its logically equivalent raised form

$$\exists y_0.\ \exists y_1.\ \forall x_0.\ \forall x_1.\ (\ x_0{=}y_0(x_0) \wedge x_1{=}y_1(x_1)\ )$$

The idea to represent this in our framework as the formula $x_0^\forall{=}y_0^\exists \wedge x_1^\forall{=}y_1^\exists$ with variable-condition $R = \{(y_0^\exists, x_1^\forall), (y_1^\exists, x_0^\forall)\}$ fails to be R-valid. Indeed, while $\{y_0^\exists{\mapsto}x_0^\forall\}$ and $\{y_1^\exists{\mapsto}x_1^\forall\}$ are existential R-substitutions, their combination $\sigma = \{y_0^\exists{\mapsto}x_0^\forall,\ y_1^\exists{\mapsto}x_1^\forall\}$ is no R-substitution:

$$\begin{tikzcd}
y_0^\exists \arrow[drr, "R"' near start] & & x_0^\forall \arrow[ll, "U_\sigma"'] \\
y_1^\exists \arrow[urr, "R" near start] & & x_1^\forall \arrow[ll, "U_\sigma"']
\end{tikzcd}$$

Now, if you want to turn this wrong representation into a proper one, you have to use the notions from the weak version of Wirth (1998) instead. Reformulated according to the slightly different notion of a substitution used in this paper, they read:

Definition Note 15.1 (Weak Variable-Condition) (Cf. Definition 4.6)
A *variable-condition* is a subset of $\mathrm{V}_\exists \times \mathrm{V}_\forall$.

Definition Note 15.2 (Weak R-Substitution) (Cf. Definition 5.2)
Let R be a variable-condition.
σ is an *R-substitution* if σ is a substitution for that $U_\sigma \circ R$ is irreflexive.

Definition Note 15.3 (Weak σ-Update) (Cf. Definition 5.3)
Let R be a variable-condition and σ be a substitution.
The *σ-update of* R is $({}_{\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma)}\upharpoonleft\mathrm{id} \cup E_\sigma) \circ R$.

Note that for this weak version we have to pay the price that we cannot use a liberalized version of the δ-rule, which makes our proofs dependent on the order in which we eliminated quantifiers, thereby severely violating our design goal of a natural flow of information, cf. Section 2.1.

Note 16: If you nevertheless want to have re-use and permutations of free existential variables you have use the following alternative notions instead.

Definition Note 16.4 (Alternative Variable-Condition) (Cf. Definition 4.6)
A *variable-condition* is a subset of $\mathrm{V}_{\mathrm{free}} \times \mathrm{V}_\forall$.

Definition Note 16.5 (Alternative R-Substitution) (Cf. Definition 5.2)
Let R be a variable-condition. σ is an *R-substitution* if σ is a substitution for that $({}_{\mathrm{V}_\forall\cup(\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma))}\upharpoonleft\mathrm{id} \cup E_\sigma \cup U_\sigma) \circ R \ \cup\ (E_\sigma \cup U_\sigma)\upharpoonright_{\mathrm{V}_\forall}$ is wellfounded.

Definition Note 16.6 (Alternative σ-Update) (Cf. Definition 5.3)
Let R be a variable-condition and σ be a substitution.
The *σ-update of* R is $({}_{\mathrm{V}_\forall\cup(\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma))}\upharpoonleft\mathrm{id} \cup E_\sigma \cup U_\sigma) \circ R \ \cup\ (E_\sigma \cup U_\sigma)\upharpoonright_{\mathrm{V}_\forall}$.

In an implementation, substituted free existential variables should get new nodes while their old nodes lose their labels. E.g., (where we have boxed the old occurrences of the re-used free existential variables $x^\exists$ and $u^\exists$) for

$$R := \{(\boxed{x^\exists}, y^\forall),\ \ (\boxed{x^\exists}, z_0^\forall),\ \ (\boxed{x^\exists}, z_1^\forall),\ \ (\boxed{x^\exists}, z_2^\forall),\ \ (\boxed{u^\exists}, v^\forall),\ \ (w^\exists, v^\forall)\}.$$

and the existential R-substitution (in the alternative sense!)

$$\sigma := \{\boxed{x^\exists} \mapsto (u^\exists + v^\forall),\ \ \boxed{u^\exists} \mapsto x^\exists,\ \ \boxed{y^\exists} \mapsto v^\forall\}$$

we should update

$$\begin{tikzcd}
\boxed{u^\exists} \arrow[r, "R"'] & v^\forall & w^\exists \arrow[l, "R"'] \\
y^\forall & \boxed{x^\exists} \arrow[l, "R"'] \arrow[dl, "R"'] \arrow[d, "R"] \arrow[dr, "R"] & \\
z_0^\forall & z_1^\forall & z_2^\forall
\end{tikzcd}$$

first to

$$\begin{tikzcd}
 & & \boxed{y^\exists} & \\
x^\exists \arrow[r, "E_\sigma"'] & \boxed{u^\exists} \arrow[r, "R"'] & v^\forall \arrow[u, "U_\sigma"'] \arrow[d, "U_\sigma"'] & w^\exists \arrow[l, "R"'] \\
 & y^\forall & \boxed{x^\exists} \arrow[l, "R"'] \arrow[dl, "R"'] \arrow[d, "R"] \arrow[dr, "R"] & u^\exists \arrow[l, "E_\sigma"'] \\
 & z_0^\forall & z_1^\forall & z_2^\forall
\end{tikzcd}$$

and then to

$$\begin{tikzcd}
x^\exists \arrow[r] & \bullet \arrow[r] & v^\forall \arrow[d] & w^\exists \arrow[l] \\
 & y^\forall & \bullet \arrow[l] \arrow[dl] \arrow[d] \arrow[dr] & u^\exists \arrow[l] \\
 & z_0^\forall & z_1^\forall & z_2^\forall
\end{tikzcd}$$

representing the σ-update of R in the alternative sense. Note that the edge from $v^\forall$ to $\boxed{y^\exists}$ has been completely removed in the last step because $y^\exists$ has no out-going R-edge. This may be an efficiency advantage over the non-alternative version, cf. also Section 15. Therefore, we have spared no efforts and included the treatment in case of the alternative definitions of R-substitution and σ-update in all our proofs!

Note 17: The other reason is that we do not have to insert $R\langle\!\{z^{\forall}\}\!\rangle \times \{y^{\forall,\mathrm{s}}\}$ into the variable-condition R anymore (as was the case in Wirth (1998)) because the transitive closure now takes care of this.

Note 18: It should be pointed out that the "some π" in this definition is something we can play around with. Indeed, in Wirth (1998), Definition 5.7 (resp. Definition 4.4 in short version), we can read "each π" instead, which is just the other extreme. The reason why we prefer "some π" to "each π" here and in Wirth (2002) is that the latter results in more valid formulas (e.g. (E2) in Wirth (2002)) and makes theorem proving easier. Contrary to the former and to all semantics for Hilbert's ε in the literature, the latter frees us from considering all possible choices: We just have to pick a single one and fix it in a proof step. As the major notion here and in Wirth (2002) is not strong validity but reduction (cf. Definition 11.1), where the quantification of π must be universal no matter how we quantify in the notion of strong validity, changing the quantification of π in Definition 10.1 would only have very local consequences. Roughly speaking, in Section 10 and Section 11 only Theor. 11.2(6a) for the case of $O \neq \emptyset$ as well as Theor. 11.2(5a) become false for a different choice on the quantification of π in Definition 10.1.

Note 19: Since the definition of wellfoundedness was changed from non-termination of the reverse relation to existence of minimal elements, the soundness of this step does not require the Principle of Descente Infinie, cf. Definition 4.4, anymore.

Note 20: Note that in Wirth (1998) the (non-liberalized) δ-rules were no sub-rules of Expansion rule of the sequent calculus of Definition 13.4 because there was only one kind of free universal variables and the δ-rule had to use the same kind of free universal variables as the liberalized δ-rule, cf. the discussion on p. 25 of Wirth (1998).

Note 21: Indeed, for the alternative notions in Note 16, we get $R := \emptyset$ here because $(x_1^{\exists}, x_2^{\exists})$ gets removed just as the edge from $v^{\forall,\mathrm{s}}$ to $y^{\exists}$ in the example of Note 16.

Note 22: Indeed, for the alternative notions in Note 16, our variable-condition would still be empty.

Note 23: Or, more formally, using the lexicographic combination of Note 13 according to $w_2^{\exists}(x) := \mathsf{lex}(\mathsf{s}(0), x, 0)$ and $w_3^{\exists}(x, y) := \mathsf{lex}(\mathsf{s}(\mathsf{s}(0)), x, y)$, or even more formally we apply the existential R-substitution $\{w_2^{\exists} \mapsto \lambda x.\ \mathsf{lex}(\mathsf{s}(0), x, 0),\ \ w_3^{\exists} \mapsto \lambda x, y.\ \mathsf{lex}(\mathsf{s}(\mathsf{s}(0)), x, y)\}$.

Note 24: If it wellfoundedness or termination were a first-order property, the first-order theory of the Peano algebra of natural numbers would be first-order axiomatizable and enumerable, but it is not even arithmetically definable, cf. e.g. Enderton (1973), p. 228.

Note 25: There is one disadvantage, however, of the liberalized δ-rule compared to the non-liberalized δ-rule. Sometimes the liberalized δ-rule results in a bigger variable-condition than the non-liberalized one because the liberalized one additionally introduces dependencies from the weak free universal variables of the principle formula, which are necessary for lemma and induction hypothesis application. One consequence of this is that simplification becomes more difficult: E.g., in item 2 at the end of Section 4.1 we *safely* removed the literal $x^{\forall,\mathrm{w}} \neq \mathsf{s}(y^{\forall,\mathrm{w}})$ from $x^{\forall,\mathrm{w}} \neq \mathsf{s}(y^{\forall,\mathrm{w}}),\ 0 + \mathsf{s}(y^{\forall,\mathrm{w}}) = \mathsf{s}(y^{\forall,\mathrm{w}});\ w_1^{\exists}(\mathsf{s}(y^{\forall,\mathrm{w}}))$ because $x^{\forall,\mathrm{w}}$ was in solved form in this sequent, cf. Note 12. If we had applied the *liberalized* δ-rule instead, we would have got $x^{\forall,\mathrm{w}}{\neq}\mathsf{s}(y^{\forall,\mathrm{s}}),\ \ \mathsf{0}+\mathsf{s}(y^{\forall,\mathrm{s}})=\mathsf{s}(y^{\forall,\mathrm{s}});\ \ w_1^{\exists}(\mathsf{s}(y^{\forall,\mathrm{s}}))$, where $x^{\forall,\mathrm{w}}$ is not in solved form because this sequent contains the strong free universal variable $y^{\forall,\mathrm{s}}$ which is not in $R^{+}\langle\{x^{\forall,\mathrm{w}}\}\rangle$. Moreover, we cannot extend the variable-condition R s.t. $R^{+}\langle\{x^{\forall,\mathrm{w}}\}\rangle$ contains $y^{\forall,\mathrm{s}}$ because the liberalized δ-rule has introduced the dependency $(x^{\forall,\mathrm{w}},y^{\forall,\mathrm{s}})$ into R, so that R would become cyclic. Note that $y^{\forall,\mathrm{s}}$ stands for $\varepsilon y.\ (x^{\forall,\mathrm{w}}{=}\mathsf{s}(y))$, which means that $x^{\forall,\mathrm{w}}$ still occurs hidden the latter sequent. Indeed, under the variable-condition $R:=\{(x^{\forall,\mathrm{w}},y^{\forall,\mathrm{s}})\}$, the choice-condition $C:=\{(y^{\forall,\mathrm{s}},(x^{\forall,\mathrm{w}}{=}\mathsf{s}(y^{\forall,\mathrm{s}})))\}$, and ($\mathsf{nat1}$) from Section 4.1, the removal of $x^{\forall,\mathrm{w}}{\neq}\mathsf{s}(y^{\forall,\mathrm{s}})$ from $x^{\forall,\mathrm{w}}{\neq}\mathsf{s}(y^{\forall,\mathrm{s}}),\ y^{\forall,\mathrm{s}}{\neq}\mathsf{0};\ \ldots$ is not safe in the sense of Definition 13.9; to wit, let $\mathcal{A}$ have the universe $\{+,-\}\times\mathbf{N}$ with $\mathsf{s}^{\mathcal{A}}(+,n):=(+,n{+}1)$, $\mathsf{s}^{\mathcal{A}}(-,n{+}1):=(-,n)$, $\mathsf{s}^{\mathcal{A}}(-,0):=(+,1)$, and $\mathsf{0}^{\mathcal{A}}:=(+,0)$, and set

$$\pi(y^{\forall,\mathrm{s}})(\tau):=\left\{\begin{array}{ll}(+,0) & \text{if } \tau(x^{\forall,\mathrm{w}}){=}(+,0)\\ (-,0) & \text{if } \tau(x^{\forall,\mathrm{w}}){=}(+,1)\\ (+,n{+}1) & \text{if } \tau(x^{\forall,\mathrm{w}}){=}(+,n{+}2)\\ (-,n{+}1) & \text{if } \tau(x^{\forall,\mathrm{w}}){=}(-,n)\end{array}\right\},$$

which is compatible with (C,R) and makes $x^{\forall,\mathrm{w}}{\neq}\mathsf{s}(y^{\forall,\mathrm{s}}),\ y^{\forall,\mathrm{s}}{\neq}\mathsf{0}$ valid, but not $y^{\forall,\mathrm{s}}{\neq}\mathsf{0}$.
There is, however, a general way to overcome this shortcoming for constructive domains. For our special case of natural numbers it looks as follows: When we add the axiom ($\mathsf{nat2}$) or ($\mathsf{nat3}$) (which together with ($\mathsf{nat1}$) is strictly stronger than ($\mathsf{nat2}$)) from below, then the removal of $x^{\forall,\mathrm{w}}{\neq}\mathsf{s}(y^{\forall,\mathrm{s}})$ from $x^{\forall,\mathrm{w}}{\neq}\mathsf{s}(y^{\forall,\mathrm{s}}),\ \Gamma;\ \ldots$ is always safe because the image of the predecessor *function* on the universe without $\mathsf{0}^{\mathcal{A}}$ is the whole universe and if Γ is false for $\tau(x^{\forall,\mathrm{w}})=\mathsf{0}^{\mathcal{A}}$ then $x^{\forall,\mathrm{w}}{\neq}\mathsf{s}(y^{\forall,\mathrm{s}}),\ \Gamma$ is false for the τ' which differs from τ in $\tau'(x^{\forall,\mathrm{w}})=\mathsf{s}^{\mathcal{A}}(\pi(y^{\forall,\mathrm{s}})(\tau))$ because then $\pi(y^{\forall,\mathrm{s}})(\tau')=\pi(y^{\forall,\mathrm{s}})(\tau)$.

($\mathsf{nat2}$) $\quad\forall x,y.\ \big(\ \mathsf{s}(x){=}\mathsf{s}(y)\Rightarrow x{=}y\ \big)$

($\mathsf{nat3}$) $\quad\forall x,y.\ \big(\ x{\longrightarrow}y\Leftrightarrow x{=}\mathsf{s}(y)\ \big)\quad\wedge\quad{\longrightarrow^{-1}}$ is wellfounded

Note 26: In terms of Hilbert's ε-operator, this asymmetry can be understood from the argumentation of Nonnengart (1996), which, for some new variable $z\in\mathrm{V}_{\mathrm{bound}}$ and t denoting the term $\varepsilon z.\ (\neg A\{x{\mapsto}z\}\wedge(A\vee x{=}z))$, employs the logical equivalence of $\forall x.\ (A{\vee}B)$ with $\forall x.\ A\vee\forall x.\ (B\{x{\mapsto}t\})$ and then the logical equivalence of $\forall x.\ A$ with $\exists x.\ (A\{x{\mapsto}t\})$.

Note 27: These calculi were presented at the 2nd Int. Workshop on First-Order Theorem Proving (FTP) in Nov. 1998 in Vienna (cf. Wirth (1998)), where nobody in the audience was able to point out other work in this direction besides the unsound calculi in Kohlhase (1995) and Kohlhase (1998), cf. Section 4.7.

Note 28: For the alternative notions in Note 16, we have to replace this sentence with the following: As R' is the σ-update of R, we have

$$U_\sigma R^{*}{\upharpoonright}_{\mathrm{V}_\forall}\subseteq U_\sigma RR^{*}\cup U_\sigma{\upharpoonright}_{\mathrm{V}_\forall}=U_\sigma R({}_{\mathrm{V}_\forall}{\upharpoonleft}R)^{*}\cup U_\sigma{\upharpoonright}_{\mathrm{V}_\forall}\subseteq R'^{+},$$

the second step being due to $\mathrm{ran}(R)\subseteq\mathrm{V}_\forall$ for any alternative variable-condition R. Similarly, ${}_{\mathrm{V}_\forall}{\upharpoonleft}(R^{+})=({}_{\mathrm{V}_\forall}{\upharpoonleft}R)^{+}\subseteq R'^{+}$. Q.e.d. (Claim 1)

Note 29: For the alternative notions in Note 16, we have to replace this sentence with the following: As R' is the σ-update of R, we have

$$({}_{\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma)}{\upharpoonleft}\mathrm{id}\cup E_\sigma)R^{*}{\upharpoonright}_{\mathrm{V}_\forall}\subseteq({}_{\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma)}{\upharpoonleft}\mathrm{id}\cup E_\sigma)RR^{*}\cup{}_{\mathrm{V}_{\mathrm{free}}}{\upharpoonleft}\mathrm{id}\cup E_\sigma{\upharpoonright}_{\mathrm{V}_\forall}=$$
$$({}_{\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma)}{\upharpoonleft}\mathrm{id}\cup E_\sigma)R({}_{\mathrm{V}_\forall}{\upharpoonleft}R)^{*}\cup{}_{\mathrm{V}_{\mathrm{free}}}{\upharpoonleft}\mathrm{id}\cup E_\sigma{\upharpoonright}_{\mathrm{V}_\forall}\subseteq R'^{*},$$

the second step being due to $\mathrm{ran}(R)\subseteq\mathrm{V}_\forall$ for any alternative variable-condition R. Q.e.d. (Claim 2)

Note 30: For the alternative notions in Note 16, we have to deviate here in the following way: Moreover, as R' is the σ-update of R, we have

$$R' = ({}_{\mathrm{V}_\forall\cup(\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma))}\rceil\mathrm{id} \cup E_\sigma \cup U_\sigma)R \cup (E_\sigma \cup U_\sigma)\upharpoonright_{\mathrm{V}_\forall}.$$

As $(R' \cup S_{e'})^+$ is a wellfounded ordering, so is its subset

$$({}_{\mathrm{V}_\forall\cup(\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma))}\rceil R \cup S_{e'}\upharpoonright_{\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma)} \cup S_{e'}E_\sigma\upharpoonright_{\mathrm{V}_\exists}R \cup U_\sigma\upharpoonright_{\mathrm{V}_\exists}R)^+.$$

The alternative version of a variable-condition guarantees $\mathrm{ran}(R) \subseteq \mathrm{V}_\forall$. Thus, additional steps with ${}_{\mathrm{V}_\exists\cap\mathrm{dom}(\sigma)}\rceil R$ must cause immediate termination; i.e.

$$(R \cup S_{e'}\upharpoonright_{\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma)} \cup S_{e'}E_\sigma\upharpoonright_{\mathrm{V}_\exists}R \cup U_\sigma\upharpoonright_{\mathrm{V}_\exists}R)^+$$

is a wellfounded ordering, too. As $\mathrm{ran}(E_\sigma\upharpoonright_{\mathrm{V}_\exists} \cup U_\sigma\upharpoonright_{\mathrm{V}_\exists}) \subseteq \mathrm{V}_\exists$ and $\mathrm{dom}(S_{e'} \cup U_\sigma) \subseteq \mathrm{V}_\forall$

$$(R \cup S_{e'}\upharpoonright_{\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma)} \cup S_{e'}E_\sigma\upharpoonright_{\mathrm{V}_\exists} \cup U_\sigma\upharpoonright_{\mathrm{V}_\exists})^+$$

is a wellfounded ordering, which is equal to $(R \cup S_e)^+$ by definition of S_e. Q.e.d. (Claim 3)

Note 31: For the alternative notions in Note 16, we have to deviate here in the following way: Moreover, as R' is the σ-update of R, we have

$$R' = ({}_{\mathrm{V}_\forall\cup(\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma))}\rceil\mathrm{id} \cup E_\sigma \cup U_\sigma)R \cup (E_\sigma \cup U_\sigma)\upharpoonright_{\mathrm{V}_\forall}.$$

As $(R' \cup S_{e'} \cup S_{\pi'})^+$ is a wellfounded ordering, so is its subset

$$\left(\begin{array}{l} {}_{\mathrm{V}_\forall\cup(\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma))}\rceil R \cup (S_{\pi'} \cup {}_{\mathrm{V}_{\forall,\mathrm{w}}}\rceil\mathrm{id})(S_{e'}({}_{\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma)}\rceil\mathrm{id} \cup E_\sigma\upharpoonright_{\mathrm{V}_\exists}) \cup U_\sigma\upharpoonright_{\mathrm{V}_\exists})(R \cup R') \\ \cup\ (R' \cup S_{e'} \cup S_{\pi'})^+\upharpoonright_{\mathrm{V}_\forall} \end{array}\right)^+.$$

The alternative version of a variable-condition guarantees $\mathrm{ran}(R \cup R') \subseteq \mathrm{V}_\forall$. Thus, additional steps with ${}_{\mathrm{V}_\exists\cap\mathrm{dom}(\sigma)}\rceil R$ must cause immediate termination; i.e.

$$(R \cup (S_{\pi'} \cup {}_{\mathrm{V}_{\forall,\mathrm{w}}}\rceil\mathrm{id})(S_{e'}({}_{\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma)}\rceil\mathrm{id} \cup E_\sigma\upharpoonright_{\mathrm{V}_\exists}) \cup U_\sigma\upharpoonright_{\mathrm{V}_\exists})(R \cup R') \cup (R' \cup S_{e'} \cup S_{\pi'})^+\upharpoonright_{\mathrm{V}_\forall})^+$$

is a wellfounded ordering, too. As $\mathrm{ran}(S_{e'}({}_{\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma)}\rceil\mathrm{id} \cup E_\sigma\upharpoonright_{\mathrm{V}_\exists}) \cup U_\sigma\upharpoonright_{\mathrm{V}_\exists}) \subseteq \mathrm{V}_\exists$ and $\mathrm{dom}(S_{\pi'} \cup {}_{\mathrm{V}_{\forall,\mathrm{w}}}\rceil\mathrm{id} \cup S_{e'} \cup S_{\pi'}) \subseteq \mathrm{V}_\forall$

$$(R \cup (S_{\pi'} \cup {}_{\mathrm{V}_{\forall,\mathrm{w}}}\rceil\mathrm{id})(S_{e'}({}_{\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma)}\rceil\mathrm{id} \cup E_\sigma\upharpoonright_{\mathrm{V}_\exists}) \cup U_\sigma\upharpoonright_{\mathrm{V}_\exists}) \cup (R' \cup S_{e'} \cup S_{\pi'})^+\upharpoonright_{\mathrm{V}_\forall})^+$$

is a wellfounded ordering, which is equal to $(R \cup S_e \cup (R' \cup S_{e'} \cup S_{\pi'})^+\upharpoonright_{\mathrm{V}_\forall})^+$ by definition of S_e. Q.e.d. (Claim 5)

Note 32: For the alternative notions in Note 16, we have to deviate here in the following way: As R' is the σ-update of R, we have

$$R' = ({}_{\mathrm{V}_\forall\cup(\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma))}\rceil\mathrm{id} \cup E_\sigma \cup U_\sigma)\circ R \cup (E_\sigma \cup U_\sigma)\upharpoonright_{\mathrm{V}_\forall}.$$

As the alternative version of a variable-condition guarantees $\mathrm{ran}(R) \subseteq \mathrm{V}_\forall$ and $\lhd$ is transitive, we have ${}_{\mathrm{V}_\forall}\rceil(R^+) \subseteq \lhd$ and $({}_{\mathrm{V}_\forall\cup(\mathrm{V}_\exists\setminus\mathrm{dom}(\sigma))}\rceil\mathrm{id} \cup E_\sigma \cup U_\sigma)\circ R^+ \subseteq \lhd$. The latter implies $S_e\circ R^+ \subseteq \lhd$ by (A.2.1). Q.e.d. (Claim 2)

D References

Peter B. Andrews (1972). *General Models, Descriptions, and Choice in Type Theory.* J. Symbolic Logic **37**, pp. 385–394.

Peter B. Andrews (2002). *An Introduction to Mathematical Logic and Type Theory: To Truth Through Proof.* 2nd ed., Academic Press.

Matthias Baaz, Christian G. Fermüller (1995). *Non-elementary Speedups between Different Versions of Tableaus.* 4th TABLEAU 1995, LNAI 918, pp. 217–230, Springer.

Matthias Baaz, Uwe Egly, Christian G. Fermüller (1997). *Lean Induction Principles for Tableaus.* 6th TABLEAU 1997, LNAI 1227, pp. 62–75, Springer.

Leo Bachmair (1988). *Proof By Consistency in Equational Theories.* 3rd IEEE symposium on Logic In Computer Sci., pp. 228–233, IEEE Press.

Peter Baumgartner, Ulrich Furbach, Frieder Stolzenburg (1997). *Computing Answers with Model Elimination.* Artificial Intelligence **90**, pp. 135–176.

Bernhard Beckert, Reiner Hähnle (1998). *Analytic Tableaus.* In: Bibel & Schmitt (1998), Vol. 1, pp. 11–41.

Bernhard Beckert, Reiner Hähnle, Peter H. Schmitt (1993). *The Even More Liberalized δ-Rule in Free Variable Semantic Tableaus.* Kurt Gödel Colloquium, LNCS 713, pp. 108–119, Springer.

Wolfgang Bibel (1987). *Automated Theorem Proving.* 2nd rev. ed., Vieweg, Braunschweig.

Wolfgang Bibel, Peter H. Schmitt (eds.) (1998). *Automated Deduction — A Basis for Applications.* Kluwer.

Susanne Biundo, Birgit Hummel, Dieter Hutter, Christoph Walther (1986). *The Karlsruhe Induction Theorem Proving System.* 8th CADE 1986, LNCS 230, pp. 672–674, Springer.

Adel Bouhoula, Michaël Rusinowitch (1995). *Implicit Induction in Conditional Theories.* J. Automated Reasoning **14**, pp. 189–235, Kluwer.

Nicolas Bourbaki (1954). *Livre I — Théorie des Ensembles.* Hermann, Paris.

Robert S. Boyer, J Strother Moore (1979). *A Computational Logic.* Academic Press.

Robert S. Boyer, J Strother Moore (1988). *A Computational Logic Handbook.* Academic Press.

Alan Bundy, Andrew Stevens, Frank van Harmelen, Andrew Ireland, Alan Smaill (1993). *Rippling: A Heuristic for Guiding Inductive Proofs.* Artificial Intelligence **62**, pp. 185–253.

Ricardo Caferra, Gernot Salzer (eds.) (2000). *Automated Deduction in Classical and Non-Classical Logics.* LNAI 1761, Springer.

Domenico Cantone, Marianna Nicolosi-Asmundo (2000). *A Further and Effective Liberalization of the δ-Rule in Free Variable Semantic Tableaus.* In: Caferra & Salzer (2000), pp. 109–125.

Serenella Cerrito, Marta Cialdea (2001). *Free-Variable Tableaus for Constant-Domain Quantified Modal Logics with Rigid and Non-Rigid Designation.* 1st IJCAR 2001, LNAI 2083, pp. 137–151, Springer.

Herbert B. Enderton (1973). *A Mathematical Introduction to Logic.* 2nd printing, Academic Press.

Melvin C. Fitting (1996). *First-Order Logic and Automated Theorem Proving.* 2nd extd. ed., Springer.

Melvin C. Fitting (1999). *On Quantified Modal Logic.* Fundamenta Informaticae **39**, pp. 105–121.

Melvin C. Fitting (2002). *Types, Tableaus, and Gödel's God.* Kluwer.

Dov M. Gabbay, C. J. Hogger, J. Alan Robinson (eds.) (1993 ff.). *Handbook of Logic in Artificial Intelligence and Logic Programming.* Clarendon Press.

Gerhard Gentzen (1935). *Untersuchungen über das logische Schließen.* Mathematische Zeitschrift **39**, pp. 176-210, 405–431.

Gerhard Gentzen (1938). *Die gegenwärtige Lage in der mathematischen Grundlagenforschung – Neue Fassung des Widerspruchsfreiheitsbeweises für die reine Zahlentheorie.* Forschungen zur Logik und zur Grundlegung der exakten Wissenschaften, Folge 4, Leipzig.

Alfons Geser (1995). *A Principle of Non-Wellfounded Induction.* In: Tiziana Margaria (ed.). Kolloquium Programmiersprachen und Grundlagen der Programmierung, MIP–9519, pp. 117–124, Univ. Passau.

Martin Giese (1998). *Integriertes automatisches und interaktives Beweisen: die Kalkülebene.* Master's thesis, Univ. Karlsruhe. `http://i11www.ira.uka.de/~giese/da.ps.gz` (May 09, 2000).

Martin Giese, Wolfgang Ahrendt (1999). *Hilbert's ε-Terms in Automated Theorem Proving.* 8th TABLEAU 1999, LNAI 1617, pp. 171–185, Springer.

Kurt Gödel (1986 ff.). *Collected Works.* Solomon Feferman (ed.), Oxford Univ. Press.

Bernhard Gramlich, Wolfgang Lindner (1991). *A Guide to* UNICOM*, an Inductive Theorem Prover Based on Rewriting and Completion Techniques.* SEKI-Report SR–91–17 (SFB), Univ. Kaiserslautern. `http://agent.informatik.uni-kl.de/seki/1991/Lindner.SR-91-17.ps.gz` (May 09, 2000).

Bernhard Gramlich, Claus-Peter Wirth (1996). *Confluence of Terminating Conditional Term Rewriting Systems Revisited.* 7th RTA 1996, LNCS 1103, pp. 245–259, Springer.

Reiner Hähnle, Peter H. Schmitt (1994). *The Liberalized δ-Rule in Free Variable Semantic Tableaus.* J. Automated Reasoning **13**, pp. 211–221, Kluwer.

David Hilbert, Paul Bernays (1968/70). *Grundlagen der Mathematik.* 2nd ed., Springer.

K. Jaakko J. Hintikka (1996). *The Principles of Mathematics Revisited.* Cambridge Univ. Press.

Paul Howard, Jean E. Rubin (1998). *Consequences of the Axiom of Choice.* Math. Surv. and Monographs, Vol. 58, American Math. Society. `http://www.math.purdue.edu/~jer/Papers/conseq.html (Oct. 15, 2002).`

Dieter Hutter (1997). *Colouring Terms to Control Equational Reasoning.* J. Automated Reasoning **18**, pp. 399–442, Kluwer.

Stig Kanger (1963). *A Simplified Proof Method for Elementary Logic.* In: Siekmann & Wrightson (1983), Vol. 1, pp. 364–371.

Deepak Kapur, Hantao Zhang (1989). *An Overview of Rewrite Rule Laboratory (*RRL*).* 3rd RTA 1989, LNCS 355, pp. 559–563, Springer.

Manfred Kerber (1998). *Proof Planning: A Practical Approach to Mechanized Reasoning in Mathemantics.* In: Bibel & Schmitt (1998), Vol. 3, pp. 77–95.

Donald E. Knuth (1997 f.). *The Art of Computer Programming.* 3rd ed., Addison-Wesley.

Michaël Kohlhase (1995). *Higher-Order Tableaus.* 4th TABLEAU 1995, LNAI 918, pp. 294–309, Springer. Revised version is: Kohlhase (1998).

Michaël Kohlhase (1998). *Higher-Order Automated Theorem Proving.* In: Bibel & Schmitt (1998), Vol. 1, pp. 431–462.

Georg Kreisel (1965). *Mathematical Logic.* In: T. L. Saaty (ed.). Lectures on Modern Mathematics, Vol. III, pp. 95–195, John Wiley & Sons, New York.

Christoph Kreitz, Brigitte Pientka (2000). *Matrix-based Inductive Theorem Proving.* 9th TABLEAU 2000, LNAI 1847, pp. 294–308, Springer.

Ulrich Kühler (2000). *A Tactic-Based Inductive Theorem Prover for Data Types with Partial Operations.* Ph.D. thesis, Infix, Sankt Augustin.

Ulrich Kühler, Claus-Peter Wirth (1996). *Conditional Equational Specifications of Data Types with Partial Operations for Inductive Theorem Proving.* SEKI-Report SR–96–11, Univ. Kaiserslautern. Short version in: 8th RTA 1997, LNCS 1232, pp. 38–52, Springer. `http://ags.uni-sb.de/~cp/p/rta97/welcome.html (Aug. 05, 2001).`

Richard C.-T. Lee (1967). *A Completeness Theorem and a Computer Program for Finding Theorems Derivable from Given Axioms.* Ph.D. thesis, Univ. of California, Berkeley.

A. C. Leisenring (1969). *Mathematical Logic and Hilbert's ε-Symbol.* Gordon and Breach, New York.

Vladimir A. Lifschitz (1971). *Specialization of the Form of Deduction in the Predicate Calculus with Equality and Function Symbols.* In: Orevkov (1971), pp. 1–24.

Vladimir A. Lifschitz (1989). *What Is the Inverse Method?.* J. Automated Reasoning **5**, pp. 1–23, Kluwer.

Sergey Yu. Maslov (1971). *The Inverse Method for Establishing Deducibility for Logic Calculi.* In: Orevkov (1971), pp. 25–96.

Dale Miller (1992). *Unification under a Mixed Prefix.* J. Symbolic Computation **14**, pp. 321–358, Academic Press.

Andreas Nonnengart (1996). *Strong Skolemization.* MPI–I–96–2–010, Max Planck-Inst. für Informatik, Saarbrücken.

Vladimir P. Orevkov (1971). *The Calculi of Symbolic Logic.I.* American Mathematical Society, Providence, Rhode Island.

Peter Padawitz (1996). *Inductive Theorem Proving for Design Specifications.* J. Symbolic Computation **21**, pp. 41–99, Academic Press.

Peter Padawitz (1998). EXPANDER. *A System for Testing and Verifying Functional Logic Programs.* `http://LS5.cs.uni-dortmund.de/~peter/ExpaTex.ps.gz` `(Sept. 14, 1999).`

Michaël S. Paterson, Mark N. Wegman (1978). *Linear Unification.* J. Computer and System Sci. **16**, pp. 158–167, Academic Press.

Dag Prawitz (1960). *An Improved Proof Procedure.* In: Siekmann & Wrightson (1983), Vol. 1, pp. 159–199.

Dag Prawitz (1965). *Natural Deduction.* Almquist & Wiksells, Uppsala.

Martin Protzen (1994). *Lazy Generation of Induction Hypotheses.* 12$^{\text{th}}$ CADE 1994, LNAI 814, pp. 42–56, Springer. Long version in: Protzen (1995).

Martin Protzen (1995). *Lazy Generation of Induction Hypotheses and Patching Faulty Conjectures.* Ph.D. thesis, Infix, Sankt Augustin.

Herman Rubin, Jean E. Rubin (1985). *Equivalents of the Axiom of Choice.* Elsevier.

N. A. Shanin, G. V. Davydov, Sergey Yu. Maslov, G. E. Mints, Vladimir P. Orevkov, A. O. Slisenko (1965). *An Algorithm for a Machine Search of a Natural Logical Deduction in a Propositional Calculus.* In: Siekmann & Wrightson (1983), Vol. 1, pp. 424–483.

Wilfried Sieg, John Byrnes (1998). *Normal Natural Deduction Proofs (in classical logic).* Studia Logica **60**, pp. 67–106, Kluwer.

Jörg H. Siekmann, G. Wrightson (eds.) (1983). *Automation of Reasoning.* Springer.

Raymond M. Smullyan (1968). *First-Order Logic.* Springer.

Wayne Snyder, Jean Gallier (1989). *Higher-Order Unification Revisited: Complete Sets of Transformations.* J. Symbolic Computation **8**, pp. 101–140, Academic Press.

Balder Ten Cate, Chung-chieh Shan (2002). *Question Answering: From Partitions to Prolog.* 11$^{\text{th}}$ TABLEAU 2002, LNAI 2381, pp. 251–265, Springer.

Lincoln A. Wallen (1990). *Automated Proof Search in Non-Classical Logics.* MIT Press. Cf., however, `http://ags.uni-sb.de/~cp/p/wallen/all.txt` for some obsolete aspects of this fascinating book.

Christoph Walther (1992). *Computing Induction Axioms.* 3rd LPAR 1992, LNAI 624, pp. 381–392, Springer.

Christoph Walther (1994). *Mathematical Induction.* In: Gabbay &al. (1993 ff.), Vol. 2, pp. 127–228.

Claus-Peter Wirth (1997). *Positive/Negative-Conditional Equations: A Constructor-Based Framework for Specification and Inductive Theorem Proving.* Ph.D. thesis, Verlag Dr. Kovač, Hamburg.

Claus-Peter Wirth (1998). *Full First-Order Sequent and Tableau Calculi With Preservation of Solutions and the Liberalized δ-Rule but Without Skolemization.* Report 698/1998, FB Informatik, Univ. Dortmund. Short version in: Gernot Salzer, Ricardo Caferra (eds.). Proc. 2nd Int. Workshop on First-Order Theorem Proving (FTP'98), pp. 244–255, Tech. Univ. Vienna, 1998. Short version also in: Caferra & Salzer (2000), pp. 283–298. `http://ags.uni-sb.de/~cp/p/ftp98/welcome.html (Aug. 05, 2001).`

Claus-Peter Wirth (1999). *Full First-Order Free Variable Sequents and Tableaus in Implicit Induction.* 8th TABLEAU 1999, LNAI 1617, pp. 293–307, Springer. `http://ags.uni-sb.de/~cp/p/tab99/welcome.html (Aug. 05, 2001).`

Claus-Peter Wirth (2002). *A New Indefinite Semantics for Hilbert's epsilon.* 11th TABLEAU 2002, LNAI 2381, pp. 298–314, Springer. `http://ags.uni-sb.de/~cp/p/epsi/welcome.html (Feb. 04, 2002).`

Claus-Peter Wirth (2003). *History and Future of Implicit and Inductionless Induction: Beware the old jade and the zombie!.* Festschrift to Jörg H. Siekmann's 60th Birthday, LNAI , Springer. `http://ags.uni-sb.de/~cp/p/zombie/welcome.html (Dec. 02, 2002).`

Claus-Peter Wirth, Klaus Becker (1995). *Abstract Notions and Inference Systems for Proofs by Mathematical Induction.* 4th CTRS 1994, LNCS 968, pp. 353–373, Springer. `http://ags.uni-sb.de/~cp/p/ctrs94/welcome.html (Aug. 05, 2001).`

Claus-Peter Wirth, Bernhard Gramlich (1994a). *A Constructor-Based Approach for Positive/Negative-Conditional Equational Specifications.* J. Symbolic Computation **17**, pp. 51–90, Academic Press. `http://ags.uni-sb.de/~cp/p/jsc94/welcome.html (Aug. 05, 2001).`

Claus-Peter Wirth, Bernhard Gramlich (1994b). *On Notions of Inductive Validity for First-Order Equational Clauses.* 12th CADE 1994, LNAI 814, pp. 162–176, Springer. `http://ags.uni-sb.de/~cp/p/cade94/welcome.html (Aug. 05, 2001).`

Ludwig Wittgenstein (1939). *Lectures on the Foundations of Mathematics, Cambridge.* Cora Diamond (ed.), from the notes of R. G. Bosanquet, Norman Malcolm, Rush Rhees, and Yorick Smythies, Univ. of Chicago Press, 1989.