An Infinitary and a Cyclic Sequent Calculus for Non-Monotone Inductive Definitions

Robbe Van den Eede[1,2,*]

[1]KU Leuven, Department of Computer Science, Belgium
[2]Vrije Universiteit Brussel, Department of Computer Science, Belgium

Abstract

Inductive definitions are an important form of knowledge in mathematics and computer science. Two common techniques to prove theorems about inductive definitions are the principle of mathematical induction and the principle of infinite descent. To formalize these principles, Brotherston and Simpson introduced the sequent calculus proof systems LKID for mathematical induction, and LKID$^\omega$ and CLKID$^\omega$ for infinite descent. LKID$^\omega$ is an infinitary system, in which proofs are infinite trees, and CLKID$^\omega$ a cyclic system, in which proofs are finite graphs. However, these calculi restrict to monotone definitions, while inductive definitions are generally non-monotone. The logic FO(ID) extends classical first-order logic with non-monotone inductive definitions. In previous work, we provided a formalization of the principle of mathematical induction for non-monotone definitions by extending LKID to a sequent calculus $\mathrm{SC}_{\mathrm{FO(ID)}}$ for FO(ID). In this work, we provide a formalization of the principle of infinite descent for non-monotone definitions by extending LKID$^\omega$ and CLKID$^\omega$ to sequent calculi $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ resp. $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ for FO(ID). We extend several proof-theoretic results for LKID$^\omega$ and CLKID$^\omega$ to $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ and $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ regarding soundness, completeness, cut-elimination and the relation with $\mathrm{SC}_{\mathrm{FO(ID)}}$.

Keywords

Sequent calculus, cyclic proofs, inductive definitions, non-monotone definitions, infinite descent

1. Introduction

1.1. Non-Monotone Inductive Definitions

In the field of Knowledge Representation and Reasoning (KRR), first-order logic (FO) is commonly used as a modeling language. However, FO is incapable of expressing some elementary and useful notions such as the set of natural numbers or the transitive closure of a graph. To enhance expressivity, one can add language constructs to FO. A particularly useful language constructs is that of inductive definitions.

Inductive definitions are an important form of knowledge, specifying a wide range of notions in mathematics and computer science. A prototypical example of an inductive definition is that of the set of natural numbers, which consists of the following rules:

1. 0 is a natural number.
2. If n is a natural number, then so is its successor $s(n)$.

An inductive definition specifies how to construct the defined set through the process of *iterated rule application*, which starts from the empty set and consequently applies rules until saturation. The natural number definition is a *monotone* definition, meaning that once a rule is applicable, it remains applicable. Not all inductive definitions are monotone, as demonstrated by the following definition of the satisfaction relation $\models$ for propositional logic:

1. $\mathcal{I} \models P$ if $P \in \mathcal{I}$.
2. $\mathcal{I} \models \varphi \wedge \psi$ if $\mathcal{I} \models \varphi$ and $\mathcal{I} \models \psi$.
3. $\mathcal{I} \models \neg\varphi$ if not $\mathcal{I} \models \varphi$.

24th International Workshop on Nonmonotonic Reasoning, July 17–19, 2026, Lisbon, Portugal
*Corresponding author.
robbe.vandeneede@kuleuven.be (R. Van den Eede)
0000-0002-2579-9053 (R. Van den Eede)

(Here we view structures $\mathcal{I}$ as sets of propositional symbols P and assume formula φ to be built from propositional symbols with conjunctions $\wedge$ and negations $\neg$.) This definition is non-monotone because of the third rule. At the start of the construction process, the defined relation is empty; hence, the third rule applies to all pairs $(\mathcal{I}, \varphi)$. It is crucial, however, not to apply this rule prematurely, for the construction process may later derive $\mathcal{I} \models \varphi$, thereby invalidating the rule. For non-monotone definitions, it is important that rules are applied *safely*, i.e., when we one is certain that the condition of a rule cannot be falsified later in the process.

The logic FO(ID) [1] is an extension of FO with a language construct to express non-monotone inductive definitions. Syntactically, FO(ID) represents inductive definitions as sets of *definitional rules*. These are expressions of the form $\forall \bar{x} : P(\bar{t}) \leftarrow \varphi$, where φ can be any FO-formula. For instance, the natural number definition can be formalized through the following FO(ID)-definition:

$$\left\{ \begin{array}{l} Nat(zero) \leftarrow \top \\ \forall n : Nat(succ(n)) \leftarrow Nat(n) \end{array} \right\}$$

FO(ID) has historical roots in logic programming, as it emerged from a line of research to assign declarative meaning to logic programs with negation as non-monotone inductive definitions [2]. Hence, it is no coincidence that the rule-based FO(ID)-definitions are suitable to capture (fragments of) logic programs. Furthermore, the *well-founded semantics* of FO(ID) is strongly based on the homonymous semantics for logic programs [3]. Denecker and Vennekens [4, 5] argued that the well-founded semantics captures the construction processes behind non-monotone inductive definitions and the essential principle of safe rule application.

FO(ID) serves as a formal scientific study of inductive definitions as they occur in mathematical texts. It provides a more general account of inductive definitions than alternative logics, which commonly impose syntactic restrictions on their definitions such as *positivity* and *stratification*. Positivity forbids negation in bodies of definitions rules, yielding in particular a notion of monotone definition. Stratification allows for negation, under the condition that the definition can be split into a hierarchy of monotone definitions, in which each defined predicate depends negatively only on defined predicates of a lower level. While less restrictive than positivity, stratification still excludes many examples of non-monotone inductive definitions [6, 7]. FO(ID) does not impose any such syntactic constraints on its definitions, thereby capturing a more general class of inductive definitions.

Discarding stratification allows for *non-total* definitions, for which the constructions process terminates in a state where rules are applicable but none safely. Intuitively, non-total definitions correspond to nonsensical or paradoxical definitions. A simple example is $\{P \leftarrow \neg P\}$, which defines the propositional symbol P as its own negation. Non-total definitions are anomalies, to be avoided in practice. But they are interesting anomalies, especially from the perspective of FO(ID) as a formal scientific study of inductive definitions. Natural language permits nonsensical definitions, and FO(ID) offers tools to formally study these definitions and to distinguish them from sensible, i.e., total definitions.

1.2. Deductive Inference for Inductive Definitions

Deductive inference or *theorem proving* is the form of reasoning in mathematical proofs, in which conclusions are drawn that infallibly follow from a set of premises. Deductive inference has practical impact. In the field of formal verification, for instance, it is employed to provide formal guarantees of correctness for software and hardware systems.

As pointed out by Brotherston and Simpson [8], two common principles to prove theorems about inductive definitions are *mathematical induction* and *infinite descent*. To formalize these principles, they introduced the *sequent calculi* LKID for mathematical induction, and LKID$^\omega$ and CLKID$^\omega$ for infinite descent. A *sequent calculus* is a style of proof system developed by Gentzen [9], well-known for its theoretical elegance and goal-directed approach to theorem proving. In the *finitary* calculus LKID, proofs take the form of finite trees, in the *infinitary* calculus LKID$^\omega$, they take the form of possibly infinite trees, and in the *cyclic* calculus CLKID$^\omega$, they take the form of finite graphs. The sequent calculi by Brotherston and Simpson restrict to positive definitions. In previous work [6, 7], we extended LKID

to a sequent calculus $\mathrm{SC}_{\mathrm{FO(ID)}}$ for FO(ID), thereby formalizing the principle of mathematical induction for non-monotone inductive definitions.

1.3. Contributions

In this work, we extend LKID$^\omega$ and CLKID$^\omega$ to sequent calculi $\mathrm{SC}^\infty_{\mathrm{FO(ID)}}$ resp. $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ for FO(ID), thereby formalizing the principle of infinite descent for non-monotone inductive definitions. Besides inductive definitions, our calculi $\mathrm{SC}^\infty_{\mathrm{FO(ID)}}$ and $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ can be used to prove theorems about logic programs under the well-founded semantics. We establish several proof-theoretic results for $\mathrm{SC}^\infty_{\mathrm{FO(ID)}}$ and $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$, extending results for LKID$^\omega$ and CLKID$^\omega$:

- Both $\mathrm{SC}^\infty_{\mathrm{FO(ID)}}$ and $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ are sound.
- $\mathrm{SC}^\infty_{\mathrm{FO(ID)}}$ is complete on a broad fragment of formulae.
- The *trace condition*, which ensures soundness on infinite paths, is decidable for $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$.
- Each proof in $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ can be transformed to a proof in a structurally favorable *cycle normal form*.
- For proofs in cycle normal form, finitary alternatives to the trace condition exist.
- In $\mathrm{SC}^\infty_{\mathrm{FO(ID)}}$, the cut rule can be eliminated on a fragment of formulae with positive definitions, and its instances can be restricted on a fragment of formulae with stratified definitions.
- $\mathrm{SC}^\infty_{\mathrm{FO(ID)}}$ can prove more theorems than $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$, which can prove more theorems than $\mathrm{SC}_{\mathrm{FO(ID)}}$.

1.4. Overview

In Section 2, we introduce the syntax and semantics of FO(ID). In Section 3, we introduce the infinitary calculus $\mathrm{SC}^\infty_{\mathrm{FO(ID)}}$ and in Section 4, the cyclic calculus $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$. In Section 5, we show that $\mathrm{SC}^\infty_{\mathrm{FO(ID)}}$ is an instance of a generic infinitary sequent calculus studied by Brotherston [10]. As a consequence, we obtain the results on soundness, decidability, cycle normalization and the finitary soundness conditions. In Sections 6, 7 and 8, we establish the results on completeness, cut-elimination and the relation between the three calculi, respectively. We conclude in Section 9 and point to avenues for further research.

2. FO(ID)

Large parts of this section appear in earlier papers [2, 1, 4, 6, 7].

2.1. Syntax

The syntax of FO(ID) extends the syntax of first-order logic (FO), which is defined as usual. A *vocabulary* is a set of non-logical symbols, which are subdivided into predicate symbols and function symbols. We write $\mathsf{ar}(\sigma)$ to refer to the arity of a predicate or function symbol σ, and σ/n to indicate that σ has arity n. A *propositional symbol* is a 0-ary predicate symbol, and a *object symbol* is a 0-ary function symbol.[1] An occurrence of a object symbol x in a formula φ is *bound* if it is in the scope of a quantified subformula $\exists x : \psi$ or $\forall x : \psi$ of φ, and is *free* otherwise. The set of freely occurring object symbols in a formula φ is denoted by $\mathrm{FV}(\varphi)$. For a set of formulae Γ, we write $\mathrm{FV}(\Gamma)$ for $\cup_{\varphi\in\Gamma}\mathrm{FV}(\varphi)$. Given a term t, an object symbol x and a formula φ, we denote by $\varphi[t/x]$ the formula obtained from φ by replacing all free occurrences of x in φ by t. Given a set of formulae Γ, we write $\Gamma[t/x]$ for the set $\cup_{\varphi\in\Gamma}\varphi[t/x]$. An occurrence of a subformula ψ in a formula φ is said to be *positive* if it occurs in the scope of an even number of negation symbols $\neg$ (after unwinding all biconditionals $\Leftrightarrow$ and material implications $\Rightarrow$ in terms of $\wedge$, $\vee$ and $\neg$), and is said to be *negative* otherwise. We use the symbol $\doteq$ for syntactic equality.

[1] We do not explicitly distinguish between variables and constants. Such distinction can be made implicitly by seeing free occurrences of objects symbols in a formula as constants and bound occurrences as variables.

Definition 2.1. A *definitional rule* (or *rule* for short) is an expression of the form $\forall \bar{x} : P(\bar{t}) \leftarrow \varphi$, where $\bar{x}$ is a tuple of object symbols, P a predicate symbol, $\bar{t}$ an $\mathsf{ar}(P)$-tuple of terms, and φ an FO-formula. The atom $P(\bar{t})$ is called the *head* and the formula φ the *body* of the definitional rule. An (FO(ID)-)*definition* is a finite set Φ of definitional rules. A predicate symbol P that occurs in the head of a rule of a definition Φ is a *defined predicate* of Φ. A *defined atom* of Φ is an atom $P(\bar{t})$ where P is a defined predicate of Φ. All non-logical symbols in Φ that are not defined predicates are called *parameters* of Φ. The set of defined predicates of Φ is denoted by $\mathrm{def}(\Phi)$, the set of parameters of Φ by $\mathrm{pars}(\Phi)$, and their union by $\mathrm{sym}(\Phi)$. We extend the notions of free and bound object symbols to definitional rules and definitions.

Alternative frameworks of inductive definitions commonly restrict to *positive* or *stratified* definitions.

Definition 2.2. Let Φ be a definition. We say that Φ is *positive* if the body φ of every definitional rule $\forall \bar{x} : P(\bar{t}) \leftarrow \varphi$ in Φ is composed of atoms by conjunctions and disjunctions (and hence, contains no negations or material implications). A *stratification* of Φ is a function $\ell : \mathrm{def}(\Phi) \to \mathbb{N}$ such that for all $P, Q \in \mathrm{def}(\Phi)$ and every definitional rule $\forall \bar{x} : P(\bar{t}) \leftarrow \varphi$ in Φ:

- if Q occurs in φ, then $\ell(Q) \leq \ell(P)$;
- if Q occurs negatively in φ, then $\ell(Q) < \ell(P)$.

We say that Φ is *stratified* if there exists a stratification of Φ.

Example 2.3. The set of even numbers can be defined by the following FO(ID)-definition of a predicate $Even/1$, in terms of the constant $zero$, the function $succ/1$ and the natural number predicate $Nat/1$:

$$\left\{ \begin{array}{l} Even(zero) \leftarrow \top \\ \forall n : Even(succ(n)) \leftarrow Nat(n) \wedge \neg Even(n) \end{array} \right\}$$

There exists no stratification ℓ of this definition, as the second rule would impose $\ell(Even) < \ell(Even)$.

The *definitional implication* $\leftarrow$ should not be confused with material implication $\Rightarrow$. An important difference is that definitional rules are not assigned truth values in structures. Definitions, on the other hand, *are* assigned truth values in structures. Intuitively, a definition Φ is satisfied in a structure $\mathcal{I}$ if $\mathcal{I}$ interprets the defined predicates of Φ as specified by the rules of Φ.

FO(ID)-formulae are defined as FO-formulae, but with the addition of the following rule:

- Φ is an FO(ID)-formula is Φ is an FO(ID)-definition.

Contrary to alternative logics of inductive definitions, definitions are an explicit part of the syntax of FO(ID) and they can occur in complex formulae.

2.2. Well-Founded Semantics

The well-founded semantics was originally developed for logic programs with negation [3]. Denecker and Vennekens [4, 5] argued that the well-founded semantics captures the semantics of non-monotone inductive definitions by formalizing the *construction* or *induction process* behind inductive definitions. Due to space constraints, we only introduce the well-founded semantics for FO(ID) intuitively. The key notion in formalizing the construction process behind inductive definitions is that of *well-founded induction*. Let Φ be a definition and $\mathcal{O}$ a $\mathrm{pars}(\Phi)$-structure, which we call a Φ-*context*. A *well-founded induction* of Φ in $\mathcal{O}$ is a (possibly transfinite) sequence $\langle \mathfrak{A}_i \rangle_{0 \leq i \leq \beta}$ of *three-valued structures*. Three-valued structures assign a *truth value* to each atom, which is true (**t**), false (**f**) or unknown (**u**). The three-valued structures $\mathfrak{A}_i$ have vocabulary $\mathrm{sym}(\Phi)$ and extend $\mathcal{O}$ (when viewed as a three-valued structure). The first structure $\mathfrak{A}_0$ maps every defined atom to unknown, reflecting that at the start of the construction process, nothing is known yet about the defined predicates. For every i, $\mathfrak{A}_{i+1}$ *refines* $\mathfrak{A}_i$ by setting some unknown atoms to true or false, given that this can be derived safely from the rules of Φ. When no more refinements apply, we have reached the *well-founded model* $\mathfrak{A}_\beta$ of Φ in $\mathcal{O}$. The *well-founded satisfaction relation* is a relation between FO-structures $\mathcal{I}$ and FO(ID)-formulae φ. It is defined as the satisfaction relation $\models$ for FO, but with addition of the following rule:

Structural rules

$$\frac{}{\Gamma \vdash \Delta}\ \Gamma \cap \Delta \neq \emptyset, \text{ or } \bot \in \Gamma \text{ or } \top \in \Delta\ (\mathsf{ax}) \qquad \frac{\Gamma' \vdash \Delta'}{\Gamma \vdash \Delta}\ \Gamma' \subseteq \Gamma, \Delta' \subseteq \Delta\ (\mathsf{wk})$$

$$\frac{\Gamma \vdash \Delta}{\Gamma[t/x] \vdash \Delta[t/x]}\ (\mathsf{subst}) \qquad \frac{\Gamma \vdash \varphi, \Delta \qquad \Gamma, \varphi \vdash \Delta}{\Gamma \vdash \Delta}\ (\mathsf{cut})$$

Logical rules

$$\frac{\Gamma \vdash \varphi, \Delta}{\Gamma, \neg\varphi \vdash \Delta}\ (\neg\mathsf{L}) \qquad \frac{\Gamma, \varphi \vdash \Delta}{\Gamma \vdash \neg\varphi, \Delta}\ (\neg\mathsf{R})$$

$$\frac{\Gamma, \varphi \vdash \Delta \qquad \Gamma, \psi \vdash \Delta}{\Gamma, \varphi \vee \psi \vdash \Delta}\ (\vee\mathsf{L}) \qquad \frac{\Gamma \vdash \varphi, \psi, \Delta}{\Gamma \vdash \varphi \vee \psi, \Delta}\ (\vee\mathsf{R})$$

$$\frac{\Gamma, \varphi, \psi \vdash \Delta}{\Gamma, \varphi \wedge \psi \vdash \Delta}\ (\wedge\mathsf{L}) \qquad \frac{\Gamma \vdash \varphi, \Delta \qquad \Gamma \vdash \psi, \Delta}{\Gamma \vdash \varphi \wedge \psi, \Delta}\ (\wedge\mathsf{R})$$

$$\frac{\Gamma \vdash \varphi, \Delta \qquad \Gamma, \psi \vdash \Delta}{\Gamma, \varphi \Rightarrow \psi \vdash \Delta}\ (\Rightarrow\mathsf{L}) \qquad \frac{\Gamma, \varphi \vdash \psi, \Delta}{\Gamma \vdash \varphi \Rightarrow \psi, \Delta}\ (\Rightarrow\mathsf{R})$$

$$\frac{\Gamma, \varphi[t/x] \vdash \Delta}{\Gamma, \forall x : \varphi \vdash \Delta}\ (\forall\mathsf{L}) \qquad \frac{\Gamma \vdash \varphi, \Delta}{\Gamma \vdash \forall x : \varphi, \Delta}\ x \notin \mathrm{FV}(\Gamma \cup \Delta)\ (\forall\mathsf{R})$$

$$\frac{\Gamma, \varphi \vdash \Delta}{\Gamma, \exists x : \varphi \vdash \Delta}\ x \notin \mathrm{FV}(\Gamma \cup \Delta)\ (\exists\mathsf{L}) \qquad \frac{\Gamma \vdash \varphi[t/x], \Delta}{\Gamma \vdash \exists x : \varphi, \Delta}\ (\exists\mathsf{R})$$

$$\frac{\Gamma[s/x, t/y] \vdash \Delta[s/x, t/y]}{\Gamma[t/x, s/y], t = s \vdash \Delta[t/x, s/y]}\ (=\mathsf{L}) \qquad \frac{}{\Gamma \vdash t = t, \Delta}\ (=\mathsf{R})$$

Figure 1: Inference rules of $\mathrm{SC}_{\mathrm{FO}}$, based on [8].

- $\mathcal{I} \models_{\mathrm{wf}} \Phi$ if $\mathcal{I}|_{\mathrm{sym}(\Phi)}$ is the well-founded model of Φ in $\mathcal{I}|_{\mathrm{pars}(\Phi)}$.

An FO(ID)-formula φ is *valid* (under the well-founded semantics) if $\mathcal{I} \models_{\mathrm{wf}} \varphi$ for all structures $\mathcal{I}$.

We say that a definition Φ is *(non-)total* in a Φ-context $\mathcal{O}$ if the well-founded model of Φ in $\mathcal{O}$ is (not) two-valued. Intuitively, Φ is non-total in $\mathcal{O}$ if Φ is a nonsensical definition given $\mathcal{O}$.

3. An Infinitary Sequent Calculus for Non-Monotone Definitions

In a sequent calculus, proofs take the form of trees or graphs, in which the nodes are labeled with *sequents*. Given a logic $\mathcal{L}$, an *$\mathcal{L}$-sequent* is an expression of the form $\Gamma \vdash \Delta$, in which Γ and Δ are finite sets of $\mathcal{L}$-formulae. If these are all formulae over a vocabulary Σ, we call $\Gamma \vdash \Delta$ a, $\mathcal{L}$-sequent *over* Σ. A *sequent calculus* $\mathrm{SC}_{\mathcal{L}}$ for $\mathcal{L}$ contains *inference rules*, deriving a *conclusion* from one or more *premises*, all of which are $\mathcal{L}$-sequents. For every connective of $\mathcal{L}$, $\mathrm{SC}_{\mathcal{L}}$ has a *left* and *right* introduction rule in $\mathrm{SC}_{\mathrm{FO(ID)}}$, introducing the connective on the left resp. right side of the conclusion. For any rule instance, a formula that occurs in the conclusion but not in any premise is called an *active* formula, and a formula that appears in a premise but not in the conclusion is called an *auxiliary* formula of the rule instance. An *$\mathrm{SC}_{\mathcal{L}}$-proof* of an $\mathcal{L}$-sequent $\Gamma \vdash \Delta$ is (generally) a rooted graph labeled with $\mathcal{L}$-sequents, such that the root is labeled with $\Gamma \vdash \Delta$ and the edges reflect instances of inference rules in $\mathrm{SC}_{\mathcal{L}}$. An *$\mathrm{SC}_{\mathcal{L}}$-theorem* is an $\mathcal{L}$-sequent $\Gamma \vdash \Delta$ of which there exists an $\mathrm{SC}_{\mathcal{L}}$-proof.

Figure 1 presents the inference rules of the sequent calculus $\mathrm{SC}_{\mathrm{FO}}$ for FO. $\mathrm{SC}_{\mathrm{FO}}$ can be seen as an extension of Gentzen's sequent calculus LK [9] with introduction rules for equality.

The inference rules of the infinitary sequent calculus $\mathrm{SC}_{\mathrm{FO(ID)}}$ extend the inference rules of $\mathrm{SC}_{\mathrm{FO}}$ with a left and a right introduction rule for defined atoms. The **right introduction rule for defined atoms** is the following:

$$\frac{\Gamma, \Phi \vdash \varphi[\bar{s}/\bar{x}], \Delta}{\Gamma, \Phi \vdash P(\bar{t}[\bar{s}/\bar{x}]), \Delta}\ (\text{def R})$$

Here Φ is a definition with a rule $\forall \bar{x} : P(\bar{t}) \leftarrow \varphi$ and $\bar{s}$ is a tuple of object symbols with the same length as $\bar{x}$. Intuitively, (def R) allows deriving heads of definitional rules from bodies.

The **left introduction rule for defined atoms** in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ is a case distinction rule:[2]

$$\frac{\text{case distinctions}}{\Gamma, \Phi, P(\bar{v}) \vdash \Delta} \text{ (case)}$$

Here Φ is a definition and $P(\bar{v})$ a defined atom of Φ. The rule has a premise $\Gamma, \Phi, \bar{v} = \bar{t}[\bar{y}/\bar{x}], \varphi[\bar{y}/\bar{x}] \vdash \Delta$ for every definitional rule $\forall \bar{x} : P(\bar{t}) \leftarrow \varphi$ in Φ defining P. Here $\bar{y}$ is a tuple of object symbols with the same length as $\bar{x}$, none of which occurs freely in $\Gamma \cup \Delta$.

In both cases, we call Φ the *corresponding definition* of the rule instance.

Definition 3.1. Let Σ be a vocabulary and $\mathcal{S}$ a Σ-sequent. Let (N, E, r, s) be a labeled directed rooted tree, where N is the set of *nodes*, $E \subseteq N \times N$ the set of *edges*, $r \in N$ the *root*, and $s : N \to \mathsf{Seqs}(\Sigma)$ the *labeling function*. Here $\mathsf{Seqs}(\Sigma)$ denotes the set of Σ-sequents. Let i be a partial function sending nodes to inference rules in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$. Suppose that $s(r) = \mathcal{S}$, and that for every $n \in N$:

- if $i(n)$ is defined, then $s(n)$ is the conclusion and $\{s(m) \mid (n, m) \in E\}$ the set of premises of an instance of the rule $i(n)$;
- if $i(n)$ is undefined, then n has no children, and we call n a *bud*.

Then we call $\mathcal{D} = (N, E, r, s, i)$ an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$*-derivation* of $\mathcal{S}$ over Σ. If $\mathcal{D}$ has no buds, we call it an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$*-pre-proof* of $\mathcal{S}$ over Σ.

Note that $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-derivations may contain infinite branches. Since not every infinite branch is acceptable, we define a condition that guarantees soundness on infinite branches. Intuitively, the condition says that along any infinite branch, some definition is "unfolded" infinitely often.

Definition 3.2. Let $\mathcal{D} = (N, E, r, s, i)$ be an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-derivation of a sequent $\mathcal{S}$ and let $\pi = (n_i)_i$ be a path in $\mathcal{D}$. A *trace* τ along π is a sequence $(\tau_i)_i$ of FO-formulae such that for all i:

- if $s(n_i) = \Gamma_i \vdash \Delta_i$, then $\tau(n_i) \in \Gamma_i$ or $\tau(n_i) \in \Delta_i$;
- if $r(n_i) =$ (subst), then $\tau_i = \tau_{i+1}[t/x]$, where $[t/x]$ is the substitution associated with the instance of (subst) applied at n_i;
- if $r(n_i) =$ (=L), and $t = s$ is the equality associated with the instance of (=L), then there exists an FO-formula φ and object symbols x and y such that $\tau_i = \varphi[t/x, s/y]$ and $\tau_{i+1} = \varphi[s/x, t/y]$;
- if $r(n_i) \notin \{\text{(subst)}, \text{(=L)}\}$ and τ_i is the active formula, then τ_{i+1} is an auxiliary formula; if $r(n_i) =$ (case) with corresponding definition Φ, then i is said to be a *progression point* w.r.t. Φ;
- if $r(n_i) \notin \{\text{(subst)}, \text{(=L)}\}$ and τ_i is not the active formula, then $\tau_i = \tau_{i+1}$.

An *(infinite) branch* in $\mathcal{D}$ is an (infinite) path in $\mathcal{D}$ starting at the root r. We say that $\mathcal{D}$ satisfies the *trace condition* if for every infinite branch π in $\mathcal{D}$, there exists a definition Φ and a trace τ along some tail of π that has infinitely many progression points w.r.t. Φ. An $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$*-proof* is an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-pre-proof that satisfies the trace condition.

The trace condition entails soundness w.r.t. the well-founded semantics by appealing to the well-foundedness of the ordinals, using the notion of well-founded induction. On a high level, the reasoning goes as follows. Suppose that $\mathcal{D}$ is an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-proof of a sequent $\mathcal{S}$, but that $\mathcal{S}$ is invalid. Then there exists an infinite branch π with invalid sequents in $\mathcal{D}$. By the trace condition, there exists a definition Φ and a trace τ along a tail of π with infinitely many progression points w.r.t. Φ. Let $\mathcal{I}$ be a countermodel of $\mathcal{S}$ and $\langle \mathfrak{A}_i \rangle_{0 \leq i \leq \beta}$ a well-founded induction of Φ in $\mathcal{I}|_{\mathrm{pars}(\Phi)}$. For every trace formula τ_i, let α_i be the least ordinal α such that τ_i is true/false in $\mathfrak{A}_\alpha$. Then, from the definition of trace, $\alpha_{i+1} \leq \alpha_i$ for all i, and $\alpha_{i+1} < \alpha_i$ if i is a progression point. Since τ has infinitely many progression points w.r.t. Φ, $(\alpha_i)_i$ is an infinitely decreasing sequence of ordinals, which contradicts the well-foundedness of the ordinals.

[2]We name the rule (case) instead of (def L) to distinguish it from the left introduction rule for defined atoms of $\mathrm{SC}_{\mathrm{FO(ID)}}$.

Example 3.3. Let Φ be the definition of $Even$ from Example 2.3, and let Γ denote the Peano axioms $\forall x, y : succ(x) = succ(y) \Rightarrow x = y$ and $\forall x : succ(x) \neq zero$. Consider the sequent $\Gamma, \Phi \vdash \neg Even(succ(zero))$, which says that, given Γ and Φ, the number one is not even. Below, we display an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-derivation of this sequent, rendered in the typical proof style, using horizontal bars for rule applications. For space reasons, we denote $zero$ by 0, abbreviate the other non-logical symbols by their first letter, and we split up the derivation in two parts.

$$\dfrac{\dfrac{}{\Gamma, \Phi, s(0) = s(n), N(n) \wedge \neg E(n) \vdash s(0) = s(n)}\,(\text{ax}) \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma, \Phi, N(0) \vdash \top}\,(\text{ax})}{\Gamma, \Phi, N(0) \vdash E(0)}\,(\text{def R})}{\Gamma, \Phi, N(0), \neg E(0) \vdash}\,(\neg\text{L})}{\Gamma, \Phi, N(0) \wedge \neg E(0) \vdash}\,(\wedge\text{L})}{\Gamma, \Phi, 0 = n, N(n) \wedge \neg E(n) \vdash}\,(=\text{L})}{\Gamma, \Phi, 0 = n, s(0) = s(n), N(n) \wedge \neg E(n) \vdash}\,(\text{wk})}{\Gamma, \Phi, s(0) = s(n) \Rightarrow 0 = n, s(0) = s(n), N(n) \wedge \neg E(n) \vdash}\,(\Rightarrow\text{L})$$

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma, \Phi, s(0) = 0, \top \vdash s(0) = 0}\,(\text{ax})}{\Gamma, \Phi, s(0) = 0, \top, s(0) \neq 0 \vdash}\,(\neg\text{L})}{\Gamma, \Phi, s(0) = 0, \top \vdash}\,(\forall\text{L}) \qquad \dfrac{\Gamma, \Phi, s(0) = s(n) \Rightarrow 0 = n, s(0) = s(n), N(n) \wedge \neg E(n) \vdash}{\Gamma, \Phi, s(0) = s(n), N(n) \wedge \neg E(n) \vdash}\,(\forall\text{L})}{\dfrac{\Gamma, \Phi, E(s(0)) \vdash}{\Gamma, \Phi \vdash \neg E(s(0))}\,(\neg\text{R})}\,(\text{case})$$

Since $\mathcal{D}$ has no buds, it is an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-pre-proof. Since it has no infinite branches, $\mathcal{D}$ trivially satisfies the trace condition and hence, it is an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-proof of $\Gamma, \Phi \vdash \neg Even(succ(zero))$.

Example 3.4. Consider the definition

$$\Phi \doteq \left\{ \begin{array}{l} \forall n, m : T(n, m) \leftarrow E(n, m) \\ \forall n, m, p : T(n, p) \leftarrow T(n, m) \wedge T(m, p) \end{array} \right\},$$

defining the transitive closure T of a binary relation E. Below, we display a segment of an infinite $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-pre-proof $\mathcal{D}$ of the sequent $\Phi, T(a, b) \vdash \exists y : E(a, y)$.

$$\dfrac{\dfrac{\dfrac{}{\Phi, b = m, E(a, m) \vdash E(a, m)}\,(\text{ax})}{\Phi, b = m, E(a, m) \vdash \exists y : E(a, y)}\,(\exists\text{R})}{\Phi, a = n, b = m, E(n, m) \vdash \exists y : E(a, y)}\,(=\text{L}) \qquad \dfrac{\dfrac{\dfrac{\dfrac{\vdots}{\Phi, \underline{T(a, m)} \vdash \exists y : E(a, y)}}{\Phi, b = p, \underline{T(a, m)}, T(m, p) \vdash \exists y : E(a, y)}\,(\text{wk})}{\Phi, b = p, \underline{T(a, m) \wedge T(m, p)} \vdash \exists y : E(a, y)}\,(\wedge\text{L})}{\Phi, a = n, b = p, \underline{T(n, m) \wedge T(m, p)} \vdash \exists y : E(a, y)}\,(=\text{L})$$

$$\dfrac{}{\Phi, \underline{T(a, b)} \vdash \exists y : E(a, y)}\,(\text{case})$$

We underlined a trace along the (segment of) the single infinite branch in $\mathcal{D}$. Since this trace has infinitely many progression points w.r.t. Φ, $\mathcal{D}$ is an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-proof.

Example 3.5. Consider the definition

$$\Phi_D \doteq \left\{ \begin{array}{l} Double(zero) \leftarrow \top \\ \forall n : Double(succ(succ(n))) \leftarrow Double(n) \end{array} \right\}.$$

Let Φ_N be the definition of Nat from Section 1 and Φ_E the definition of $Even$ from Example 2.3, and let $\Gamma = \{\Phi_N, \Phi_E, \Phi_D\}$. Below, we display a segment of an infinite $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-pre-proof $\mathcal{D}$ of the sequent

$\forall n : \mathit{Even}(n) \Rightarrow \mathit{Double}(n)$. As before, we denote zero by 0, abbreviate other non-logical symbols by their first letter, and split up the derivation in two parts.

$$\dfrac{\dfrac{}{\Gamma, \top \vdash \top, D(s(0))}\,(\text{ax})}{\dfrac{\Gamma, \top \vdash E(0), D(s(0))}{\Gamma, m = 0, \top \vdash E(m), D(s(m))}\,(\text{def R})}\,(=\text{L}) \qquad \dfrac{\dfrac{\dfrac{}{\Gamma, N(p) \vdash N(p), D(p)}\,(\text{ax}) \quad \dfrac{\vdots}{\dfrac{\Gamma, \underline{E(p)} \vdash D(p)}{\Gamma, N(p), \underline{E(p)} \vdash D(p)}\,(\text{wk})}}{\Gamma, N(p) \vdash \underline{N(p)} \wedge \neg E(p), D(p)}\,(\wedge\text{R})}{\Gamma, N(p) \vdash \underline{E(s(p))}, D(p)}\,(\text{def R})}{\Gamma, N(p) \vdash \underline{E(s(p))}, D(s(s(p)))}\,(\text{def R})}{\Gamma, m = s(p), N(p) \vdash \underline{E(m)}, D(s(m))}\,(=\text{L})$$

$$\dfrac{\dfrac{\dfrac{}{\Gamma, \top \vdash \top}\,(\text{ax})}{\Gamma, \top \vdash D(0)}\,(\text{def R})}{\Gamma, n = 0, \top \vdash D(n)}\,(=\text{L}) \qquad \dfrac{\dfrac{\dfrac{\Gamma, N(m) \vdash \underline{E(m)}, D(s(m))}{\Gamma, N(m), \underline{\neg E(m)} \vdash D(s(m))}\,(\neg\text{L})}{\Gamma, n = s(m), N(m), \underline{\neg E(m)} \vdash D(n)}\,(=\text{L})}{\Gamma, n = s(m), \underline{N(m) \wedge \neg E(m)} \vdash D(n)}\,(\wedge\text{L})$$

$$\dfrac{\dfrac{\dfrac{\Gamma, \underline{E(n)} \vdash D(n)}{\Gamma \vdash \underline{E(n) \Rightarrow D(n)}}\,(\Rightarrow\text{R})}{\Gamma \vdash \underline{\forall n : E(n) \Rightarrow D(n)}}\,(\forall\text{R})}{\Gamma, N(m) \vdash \underline{E(m)}, D(s(m))}\,(\text{case})$$

We underlined a trace along the (segment of) the single infinite branch in $\mathcal{D}$. Since this trace has infinitely many progression points w.r.t. Φ_E, $\mathcal{D}$ is an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-proof.

4. A Cyclic Sequent Calculus for Non-Monotone Definitions

Many infinite proofs can be represented in a finite way. Our cyclic calculus $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ can be seen as the restriction of $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ to *regular trees*. These are trees with only finitely many distinct subtrees. It is well-known that regular trees are precisely the tree unravelings of finite graphs.

Definition 4.1. Let Σ be a vocabulary and $\mathcal{S}$ an FO(ID)-sequent over Σ. An $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$*-derivation* $\mathcal{D} = (N, E, r, s, i)$ of $\mathcal{S}$ over Σ is defined as an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-derivation, with the difference that (N, E) is a finite graph instead of a possibly infinite tree. Let $n, m \in N$. We call n a *companion* of m if $s(n) = s(m)$. A node $n \in N$ is called a *companion* of a node $n \in N$ in $\mathcal{D}$ if $n \neq m$ but $s(n) = s(m)$. A *repeat function* $\mathcal{R}$ for $\mathcal{D}$ maps every bud of $\mathcal{D}$ to a companion. An $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$*-pre-proof* $\mathcal{P}$ is a pair $(\mathcal{D}, \mathcal{R})$ of an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-derivation $\mathcal{D}$ and a repeat function $\mathcal{R}$ for $\mathcal{D}$. The *graph* $\mathcal{G}_{\mathcal{P}}$ of an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-pre-proof is obtained by identifying buds with their companions. An $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$*-proof* is an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-pre-proof for which $\mathcal{G}_{\mathcal{P}}$ satisfies the trace condition.

Example 4.2. The $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-proof $\mathcal{D}$ from Example 3.4 can easily be transformed into an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-proof, by replacing the top node of the displayed segment with the derivation

$$\dfrac{\Phi, T(a, b) \vdash \exists y : E(a, y)}{\Phi, T(a, m) \vdash \exists y : E(a, y)}\,(\text{subst})$$

and letting $\mathcal{R}$ map the upper node to its (unique) companion in $\mathcal{D}$.

In a similar way, the $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-proof $\mathcal{D}$ from Example 3.5 can be transformed into an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-proof.

$\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ and $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ allow proving non-totality of definitions Φ, by deriving sequents $\vdash \neg\Phi$. Indeed, a sequent of the form $\vdash \neg\Phi$ being valid means that Φ has no two-valued models. In other words, the well-founded model of Φ in any Φ-context must be strictly three-valued. The calculi $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ and $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ thus allow us to formally prove that a definition is nonsensical or paradoxical.

Example 4.3. Let Φ be the definition $\{P \leftarrow \neg P\}$. Consider the following $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-pre-proof $\mathcal{P}$ of the sequent $\vdash \neg\Phi$, where we represent the repeat function $\mathcal{R}$ by the annotations $(*)$ and $(\dagger)$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\Phi, \underline{P} \vdash \ (*)}{\Phi \vdash \underline{\neg P}}\,(\neg\mathrm{R})}{\Phi \vdash \underline{P} \ (\dagger)}\,(\mathrm{def\ R}) \qquad \dfrac{\dfrac{\Phi \vdash \underline{P} \ (\dagger)}{\Phi, \underline{\neg P} \vdash}\,(\neg\mathrm{L})}{\Phi, \underline{P} \vdash \ (*)}\,(\mathrm{case})}{\Phi \vdash}\,(\mathrm{cut})}{\vdash \neg\Phi}\,(\neg\mathrm{R})$$

The graph $\mathcal{G}_{\mathcal{P}}$ of $\mathcal{P}$ has two infinite branches, which share a tail. We underlined a trace along this tail that has infinitely many progression points w.r.t.Φ. Therefore, $\mathcal{P}$ is an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-proof.

Example 4.4. Let $\Phi \doteq \begin{Bmatrix} P \leftarrow \neg Q \\ Q \leftarrow \neg P \end{Bmatrix}$, and consider the following $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-pre-proof $\mathcal{P}$ of $\vdash \neg\Phi$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\Phi \vdash \underline{P} \ (*)}{\Phi, \underline{\neg P} \vdash}\,(\neg\mathrm{L})}{\Phi, \underline{Q} \vdash}\,(\mathrm{case})}{\Phi \vdash \underline{\neg Q}}\,(\neg\mathrm{R})}{\Phi \vdash \underline{P} \ (*)}\,(\mathrm{def\ R}) \qquad \dfrac{\dfrac{\dfrac{\dfrac{\Phi, \underline{P} \vdash \ (\dagger)}{\Phi \vdash \underline{\neg P}}\,(\neg\mathrm{R})}{\Phi \vdash \underline{Q}}\,(\mathrm{def\ R})}{\Phi, \underline{\neg Q} \vdash}\,(\neg\mathrm{L})}{\Phi, \underline{P} \vdash \ (\dagger)}\,(\mathrm{case})}{\Phi \vdash}\,(\mathrm{cut})}{\vdash \neg\Phi}\,(\neg\mathrm{R})$$

The graph $\mathcal{G}_{\mathcal{P}}$ of $\mathcal{P}$ has two infinite branches. We underlined a trace along tails of these branches, both with infinitely many progression points w.r.t. Φ. This shows that $\mathcal{P}$ is an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-proof.

Example 4.5. The definition $\Phi \doteq \{\forall n : E(n) \leftarrow \neg E(s(n))\}$ can be seen as an erroneous attempt to define the even numbers. Consider the following $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-pre-proof $\mathcal{P}$ of $\vdash \neg\Phi$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\Phi, \underline{E(x)} \vdash \ (*)}{\Phi, \underline{E(s(x))} \vdash}\,(\mathrm{subst})}{\Phi \vdash \underline{\neg E(s(x))}}\,(\neg\mathrm{R})}{\Phi \vdash \underline{E(x)} \ (\dagger)}\,(\mathrm{def\ R}) \qquad \dfrac{\dfrac{\Phi \vdash \underline{E(x)} \ (\dagger)}{\Phi \vdash \underline{E(s(x))}}\,(\mathrm{subst})}{\Phi, x = y, \underline{\neg E(s(y))} \vdash}\,(=\mathrm{L})}{\Phi, \underline{E(x)} \vdash \ (*)}\,(\mathrm{case})}{\vdash}\,(\mathrm{cut})}{\vdash \neg\Phi}\,(\neg\mathrm{R})$$

As before, the graph $\mathcal{G}_{\mathcal{P}}$ of $\mathcal{P}$ has two infinite branches, which share a tail. We underlined a trace along this tail with infinitely many progression points w.r.t.Φ, showing that $\mathcal{P}$ is an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-proof.

5. A Generic Infinitary Calculus

In his PhD thesis [10], Brotherston studied a generic infinitary calculus SC^{∞} for a generic logic $\mathcal{L}$. The only requirements on $\mathcal{L}$ and SC^{∞} are that: (1) $\mathcal{L}$ has a notion of *sequent*,[3] a notion of *interpretation*, and

[3]These *sequents* need not be of the form $\Gamma \vdash \Delta$ as defined earlier. They could also take the form of formulae φ, for instance.

a notion of *satisfaction relation* $\models$ between interpretations and sequents; and (2) the inference rules in SC^{∞} are of the form

$$\frac{S_1 \quad \dots \quad S_n}{S}\ (\mathrm{R})$$

for $S_1, \dots, S_n, S \in \mathrm{Seqs}$. Brotherston defined a generic notion of trace condition for SC^{∞}-pre-proofs, stating that every infinite branch must have infinitely many *progression points* in some general sense.

Our infinitary calculus $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ is an instance of the generic infinitary calculus SC^{∞}, and our cyclic calculus $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ an instance of the corresponding generic cyclic calculus $\mathrm{SC}^{\circlearrowleft}$. As a consequence, $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ and $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ inherit the properties proven by Brotherston about SC^{∞} and $\mathrm{SC}^{\circlearrowleft}$.

Theorem 5.1 (Soundness)**.** *Let $\mathcal{S}$ be an* FO(ID)*-sequent. If there exists an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-proof or an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-proof of $\mathcal{S}$, then $\mathcal{S}$ is valid.*

Theorem 5.2. *It is decidable to determine whether an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-pre-proof is an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-proof.*

Furthermore, we obtain *cycle normalization* and the existence of *trace manifolds* for $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$. (For space reasons, we only discuss these properties informally.)

An $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-(pre-)proof $\mathcal{P} = (\mathcal{D}, \mathcal{R})$ is in *cycle normal form* if for every bud n of $\mathcal{P}$, the companion $\mathcal{R}(n)$ of n is an ancestor of n in $\mathcal{D}$. *Cycle normalization* for $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ says that every $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-(pre-)proof $\mathcal{P}$ can be transformed into an *equivalent* $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-(pre-)proof $\mathcal{P}'$ in normal form; equivalent in the sense that $\mathcal{P}$ and $\mathcal{P}'$ have the same tree unfolding. Cycle normalization is valuable in proof search, as it assures that one can restrict to ancestors of buds when searching for companions in pre-proofs.[4]

Pre-proofs in cycle normal form admit notions of *trace manifold*. These provide finitary alternatives to the trace criterion, which quantifies over all infinite paths in the graph $\mathcal{G}_{\mathcal{P}}$ of a proof $\mathcal{P}$. Intuitively, a trace manifold for $\mathcal{P}$ consists of a collection of traces along finite paths in $\mathcal{G}_{\mathcal{P}}$, together with conditions that these traces can be "glued together" to form traces along all infinite paths in $\mathcal{G}_{\mathcal{P}}$. Brotherston introduced two equivalent notions of trace manifolds: one in terms of *strongly connected subgraphs* and one in terms of an *induction order* [10]. While these trace manifold conditions are more restrictive than the trace condition, they have the benefit of being finitary and more explicit.

6. Completeness

Neither of the calculi $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ and $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ is complete, as shown by the elementary counterexample $P \vdash \{P \leftarrow \top\}$. While this sequent is clearly valid, it is not provable in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$, for the simple reason that $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ has no inference rule to introduce definitions on the right side of sequents. $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ is complete on *regular* sequents, however, in which definitions only occur atomically on the left side.

Definition 6.1. We call an $\mathrm{SC}_{\mathrm{FO(ID)}}$-sequent *regular* if it is of the form $\Gamma, \Phi_1, \dots, \Phi_n \vdash \Delta$, where $\Phi_1, \dots, \Phi_n$ are definitions, and Γ and Δ sets of FO-formulae.

Theorem 6.2 (Completeness)**.** *Let $\mathcal{S}$ be a regular $\mathrm{SC}_{\mathrm{FO(ID)}}$-sequent. If $\mathcal{S}$ is valid, then there exists an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-proof of $\mathcal{S}$.*

Our completeness proof is based on the proof of cut-free completeness LKID^{ω} by Brotherston and Simpson [8]. It proceeds along the following lines:

- We fix a regular sequent $\mathcal{S}$ and construct a pre-proof of $\mathcal{S}$ called the *search tree* $\mathcal{D}_{\omega}$ for $\mathcal{S}$. Intuitively, $\mathcal{D}_{\omega}$ corresponds to an exhaustive search for a proof of $\mathcal{S}$. Our notion of search tree differs from that of Brotherston and Simpson, since we incorporate the cut rule in it.
- If $\mathcal{D}_{\omega}$ is a proof of $\mathcal{S}$, then we are done. If not, there must be a branch in $\mathcal{D}_{\omega}$ along which no trace exists with infinitely many progression points w.r.t. a definition Φ. We fix such branch and refer to it as the *untraceable branch* π^{u}.

[4] On the other hand, proofs in cycle normal form may in worst case be exponentially larger than the smallest proof of the same sequent that is not in cycle normal form [10].

- Based on π^{u}, we construct a structure $\mathcal{I}_\omega$ that we show the be a countermodel of $\mathcal{S}$.
- In conclusion, we have shown that every regular sequent either has a proof or a countermodel.

The most innovative part of the proof goes in showing that $\mathcal{I}_\omega$ is a countermodel of $\mathcal{S}$, and more specifically that $\mathcal{I}_\omega$ satisfies the definitions $\Phi_1, \dots, \Phi_n$ in $\mathcal{S} \doteq \Gamma, \Phi_1, \dots, \Phi_n \vdash \Delta$. We do so by showing that for every defined atom $P(\bar{t})$ of every definition Φ_i, $P(\bar{t})$ is true in the well-founded model of Φ_i in $\mathcal{I}_\omega|_{\mathrm{pars}(\Phi_i)}$ iff $P(\bar{t})$ is true in $\mathcal{I}_\omega$. Intuitively, the untraceable branch π^{u} provides enough structural information to extract a derivation of the truth of $P(\bar{t})$ in this well-founded model in case $\mathcal{I}_\omega \models P(\bar{t})$ and, using the untraceability of π^{u}, the absence of such a reason in case $\mathcal{I}_\omega \not\models P(\bar{t})$.

On a technical level, we prove this by employing the semantic framework of *justification theory* [11, 12], which characterizes the well-founded semantics through tree-like (or graph-like in some more recent versions [13, 14]) objects called *justifications*. Justifications formalize the construction process behind non-monotone inductive definitions, and as such, they provide an alternative to the well-founded inductions from Section 2.2.

As a consequence of Gödel's first incompleteness theorem, the set of regular $\mathrm{SC}_{\mathrm{FO(ID)}}$-theorems is not recursively enumerable. Since $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ is finitary, the set of (regular) $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-theorems is recursively enumerable, and hence, by Gödel's first incompleteness theorem, an analogous completeness result cannot hold for $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$.

7. Cut-Elimination

Brotherston and Simpson showed that LKID^ω is *cut-free* complete, meaning that every valid sequent admits an LKID^ω-proof without (cut). Together with soundness, this implies *cut-elimination* for LKID^ω, which says that every provable sequent in LKID^ω is also provable in LKID^ω without (cut). Cut-elimination is a fundamental result in proof theory, originally shown by Gentzen for his sequent calculi LK and LJ for classical resp. intuitionistic first-order logic [9]. Since the cut rule can intuitively be seen as the application of a lemma, cut-elimination intuitively says that every theorem can be proven without the need to introduce lemmas. Cut-elimination is valuable for proof search, as the *cut formula* φ in (cut), which formalizes the lemma, can be any FO-formula.

Cut-elimination does *not* hold for $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ or $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$. This is shown by the proof in Example 4.3, for instance, as a simple inspection of the inference rules of $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ and $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ reveals that the sequent $\Phi \vdash$ cannot be proven with any other rule than (cut). This sequent is also a counterexample of cut-elimination for $\mathrm{SC}_{\mathrm{FO(ID)}}$ [7], even though cut-elimination holds for LKID [8]. Slightly extending Brotherston and Simpson's proof of cut-elimination for LKID, however, we have obtained cut-elimination for $\mathrm{SC}_{\mathrm{FO(ID)}}$ on a fragment of regular sequents with positive definitions [7]. The same holds for $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$.

Definition 7.1. Let $\Gamma, \Phi_1, \dots, \Phi_n \vdash \Delta$ be a regular sequent. We say that $\Gamma, \Phi_1, \dots, \Phi_n \vdash \Delta$ is *canonical* if for all $i, j \in \{1, \dots, n\}$:

1. the body of each definitional rule of Φ_i is composed of literals by conjunctions and disjunctions;
2. if $i \neq j$, then $\mathrm{def}(\Phi_i) \cap \mathrm{def}(\Phi_j) = \emptyset$;
3. if $j < i$, then $\mathrm{def}(\Phi_i) \cap \mathrm{pars}(\Phi_j) = \emptyset$.

We call a sequent $\mathcal{S}$ *cut-free provable* in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ if there exists an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-proof of $\mathcal{S}$ without (cut).

Theorem 7.2 (Cut-Free Completeness)**.** *Let $\Gamma, \Phi_1, \dots, \Phi_n \vdash \Delta$ be a canonical sequent with positive definitions $\Phi_1, \dots, \Phi_n$. If $\Gamma, \Phi_1, \dots, \Phi_n \vdash \Delta$ is valid, then it is cut-free provable in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$.*

Theorem 7.3 (Cut-Elimination)**.** *Let $\Gamma, \Phi_1, \dots, \Phi_n \vdash \Delta$ be a canonical sequent with positive definitions $\Phi_1, \dots, \Phi_n$. If $\Gamma, \Phi_1, \dots, \Phi_n \vdash \Delta$ is provable in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$, then it is cut-free provable in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$.*

For $\mathrm{SC}_{\mathrm{FO(ID)}}$, we showed a *cut-restriction* result for canonical sequents with stratified definitions, saying that sequents in this form can be proven with cuts of a very specific form [7]. By the same reasoning as in [7], this result extends to $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$.

Definition 7.4. We say that a regular sequent $\Gamma, \Phi_1, \ldots, \Phi_n \vdash \Delta$ is *EC-provable* (EC standing for elementary cut) in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ if there exists an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-proof of $\Gamma, \Phi_1, \ldots, \Phi_n \vdash \Delta$ in which the cut formula in every application of (cut) is of the form $\forall \bar{y} : Q(\bar{y}) \vee \neg Q(\bar{y})$, such that Q appears negatively in the body of a rule of a definition Φ_i, and $\bar{y}$ is a fixed $\mathsf{ar}(Q)$-tuple of object symbols.

Theorem 7.5 (EC-Completeness)**.** *Let $\Gamma, \Phi_1, \ldots, \Phi_n \vdash \Delta$ be a canonical sequent with stratified definitions $\Phi_1, \ldots, \Phi_n$. If $\Gamma, \Phi_1, \ldots, \Phi_n \vdash \Delta$ is valid, then it is EC-provable in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$.*

Theorem 7.6 (Cut-Restriction)**.** *Let $\Gamma, \Phi_1, \ldots, \Phi_n \vdash \Delta$ be a canonical sequent with stratified definitions $\Phi_1, \ldots, \Phi_n$. If $\Gamma, \Phi_1, \ldots, \Phi_n \vdash \Delta$ is provable in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$, then it is EC-provable in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$.*

Note that Theorems 7.5 and 7.6 generalize Theorems 7.2 resp. 7.3, as for positive definitions, EC-provability in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ comes down to provability in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$.

Cut-restriction is valuable for proof search, as it narrows the required cut formulae down to a finite set, bounded by the defined predicates of the definitions $\Phi_1, \ldots, \Phi_n$.

It is currently unknown whether Theorem 7.6 can be strengthened to full cut-elimination. Cut-elimination may thus hold for the broader fragment of canonical sequents with stratified definitions and, as far as we know, even for total definitions.

A similar result as 7.3 does not hold for $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$, since Oda et al. disproved cut-elimination for CLKID$^{\omega}$ [15], confirming a conjecture by Brotherston [10].

8. Relation between $\mathrm{SC}_{\mathrm{FO(ID)}}$, $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ and $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$

Since every $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-proof unfolds into an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-proof, every $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-theorem is also an $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-theorem. Conversely, not every $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$-theorem is an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-theorem, as $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ is complete on regular sequents but $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ is not.

Brotherston and Simpson showed that every LKID-theorem is also a CLKID$^{\omega}$-theorem, and they conjectured the converse to hold as well [8]. Their conjecture was later disproven by Berardi and Tatsuta [16], whose counterexample involves only the natural number definition from Section 1. This counterexample extends to show that not every $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-theorem is an $\mathrm{SC}_{\mathrm{FO(ID)}}$-theorem. An additional counterexample of the latter is the sequent $\vdash \neg\Phi$ from Example 4.4. This sequent is not provable in $\mathrm{SC}_{\mathrm{FO(ID)}}$, as $\mathrm{SC}_{\mathrm{FO(ID)}}$ is sound w.r.t. the *stable (model) semantics* (which also originates from Logic Programming [17]) [7], and $\vdash \neg\Phi$ is not valid under this semantics.

We show that every $\mathrm{SC}_{\mathrm{FO(ID)}}$-theorem is an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-theorem, extending Brotherston and Simpson's proof of the fact that every LKID-theorem is a CLKID$^{\omega}$-theorem [8]. The crux lies in showing that every instance of the *induction rule* (ind), which is the left introduction rule for defined atoms in $\mathrm{SC}_{\mathrm{FO(ID)}}$ [6, 7], is derivable in $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$.

Lemma 8.1. *For every instance of* (ind) *with conclusion $\mathcal{S}$ and premises $\mathcal{S}_1, \ldots, \mathcal{S}_n$, there exists an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-derivation of $\mathcal{S}$ satisfying the global trace condition, the buds of which are labeled by $\mathcal{S}_1, \ldots, \mathcal{S}_n$.*

Theorem 8.2. *Every $\mathrm{SC}_{\mathrm{FO(ID)}}$-theorem is an $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$-theorem.*

The extensions of LKID$^{\omega}$ and CLKID$^{\omega}$ to $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ resp. $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ are arguably more elegant than the extension of LKID to $\mathrm{SC}_{\mathrm{FO(ID)}}$. To extend the induction rule (ind) to non-monotone definitions, we introduced an asymmetry between positive and negative occurrences of defined predicates, deviating further from the informal principle of mathematical induction as commonly used in pen-and-paper proofs. The case distinction rule (case) in LKID$^{\omega}$ and CLKID$^{\omega}$, on the other hand, is essentially[5] identical to the corresponding rule in LKID$^{\omega}$ and CLKID$^{\omega}$. To accommodate for non-monotone definitions, we only needed to extend the trace condition. While our notion of trace condition is more complicated than the Brotherston and Simpson's, the underlying intuition of "unfolding" a definition infinitely often along infinite paths remains.

[5] Since definitions are implicit in the logic of Brotherston and Simpson, they do not appear in their inference rules.

$\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ and $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ furthermore have the advantage of being more tailored towards the well-founded semantics than $\mathrm{SC}_{\mathrm{FO(ID)}}$. Indeed, since $\mathrm{SC}_{\mathrm{FO(ID)}}$ is sound w.r.t. the stable semantics, it cannot prove sequents that are invalid under the stable semantics, such as the sequent from Example 4.4. $\mathrm{SC}_{\mathrm{FO(ID)}}$, on the other hand, has the advantage of being more broadly applicable, as it can also be used to prove theorems about logic programs under the stable semantics.

9. Conclusion

In this paper, we provided a formalization of the principle of infinite descent for non-monotone inductive definitions by extending the infinitary sequent calculus LKID^{ω} and the cyclic sequent calculus CLKID^{ω} by Brotherston and Simpson [8] to the sequent calculi $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ resp. $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ for FO(ID). Furthermore, we extended various results about LKID^{ω} and CLKID^{ω} to $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ and $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$, thereby providing a solid proof-theoretic evaluation of our sequent calculi.

This work opens several avenues for future research:

- Our cut-elimination result for $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ can potentially be extended to canonical sequents with stratified or even total definitions.
- The counterexample $P \vdash \{P \leftarrow \top\}$ from Section 6 for completeness of $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ on non-regular sequents relies on the fact that $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ has no right introduction rule for definitions. Such a rule does occur in the work of Hou Ping et al. [18, 19],[6] raising the question whether including this rule in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ entails completeness on general sequents.
- In Section 4, we saw that $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ and $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ can be used to prove non-totality of definitions (as is the case for $\mathrm{SC}_{\mathrm{FO(ID)}}$ [6, 7]). We cannot prove totality of definitions, however, a first obstacle being that we cannot express totality in the syntax of FO(ID). Therefore, we may investigate how to extend the logic FO(ID) and its proof systems to enable proofs of totality of definitions.
- Due to their structural similarities, we expected justification theory to play a role in the soundness criterion for pre-proofs in $\mathrm{SC}^{\infty}_{\mathrm{FO(ID)}}$ and $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$. As the trace condition does not involve justification theory, the question remains whether there exist notions of infinitary and cyclic calculus that are explicitly based on justification theory. Since justification theory has grown into a uniform framework for capturing the semantics of several non-monotone logics (among which the stable semantics) [13, 14], such notions could provide calculi for all these logics.
- Since the formalization of mathematical proofs is an important motivation of our work, it would be sensible to develop *natural deduction* counterparts to our sequent calculi. This is another style of proof system by Gentzen, aiming to come "as close as possible to actual reasoning" [9].
- We could extend our calculi to extensions of FO(ID) with additional language constructs such as types (or sorts), aggregates, partial functions and modal operators.
- We could implement an automated theorem prover for $\mathrm{SC}^{\circlearrowleft}_{\mathrm{FO(ID)}}$ by instantiating the generic cyclic theorem prover Cyclist by Brotherston et al. [20].

Acknowledgments

This work was supported by Fonds Wetenschappelijk Onderzoek Vlaanderen (FWO Flanders) under the project 11A2R26N.

The author wishes to thank Marc Denecker and Bart Bogaerts for the valuable feedback they provided during discussions about this paper.

Declaration on Generative AI

During the preparation of this work, the author used Claude Sonnet 4.6 in order to: Improve writing style. After using this tool, the author reviewed and edited the content as needed and takes full responsibility

[6] Notably, this rule is only sound for total definitions.

for the publication's content.

References

[1] M. Denecker, E. Ternovska, A logic of nonmonotone inductive definitions, ACM Trans. Comput. Log. 9 (2008) 14:1–14:52.

[2] M. Denecker, M. Bruynooghe, V. Marek, Logic programming revisited: Logic programs as inductive definitions, ACM Trans. Comput. Log. 2 (2001) 623–654.

[3] A. Van Gelder, K. A. Ross, J. S. Schlipf, The well-founded semantics for general logic programs, J. ACM 38 (1991) 620–650.

[4] M. Denecker, J. Vennekens, The well-founded semantics is the principle of inductive definition, revisited, in: C. Baral, G. De Giacomo, T. Eiter (Eds.), KR, AAAI Press, 2014, pp. 1–10.

[5] M. Denecker, The well-founded semantics is the principle of inductive definition, in: J. Dix, L. F. del Cerro, U. Furbach (Eds.), JELIA, volume 1489 of *LNCS*, Springer, 1998, pp. 1–16.

[6] R. Van den Eede, R. Van Biervliet, M. Denecker, A sequent calculus for generalized inductive definitions, in: C. Dodaro, G. Gupta, M. V. Martinez (Eds.), LPNMR, Springer Nature Switzerland, 2024, pp. 30–42.

[7] R. Van den Eede, M. Denecker, A sequent calculus for general inductive definitions, 2026. arXiv:2604.19382.

[8] J. Brotherston, A. Simpson, Sequent calculi for induction and infinite descent, Journal of logic and computation 21 (2011) 1177–1216.

[9] G. Gentzen, Untersuchungen Über das logische schliessen. i., Mathematische Zeitschrift 39 (1935) 176–210, 405–431. Reproduced with English translation in [21], 68–131.

[10] J. Brotherston, Sequent calculus proof systems for inductive definitions, Ph.D. thesis, University of Edinburgh. College of Science and Engineering. School of Informatics, 2006.

[11] M. Denecker, D. De Schreye, Justification semantics: A unifying framework for the semantics of logic programs, in: L. M. Pereira, A. Nerode (Eds.), LPNMR, MIT Press, 1993, pp. 365–379.

[12] M. Denecker, Knowledge representation and reasoning in incomplete logic programming, Ph.D. thesis, K.U.Leuven, Leuven, Belgium, 1993.

[13] M. Denecker, G. Brewka, H. Strass, A formal theory of justifications, in: F. Calimeri, G. Ianni, M. Truszczyński (Eds.), Logic Programming and Nonmonotonic Reasoning - 13th International Conference, LPNMR 2015, Lexington, KY, USA, September 27-30, 2015. Proceedings, volume 9345 of *Lecture Notes in Computer Science*, Springer, 2015, pp. 250–264.

[14] S. Marynissen, Advances in Justification Theory, Ph.D. thesis, Department of Computer Science, KU Leuven, 2022. Denecker, Marc and Bart Bogaerts (supervisors).

[15] Y. Oda, J. Brotherston, M. Tatsuta, The failure of cut-elimination in cyclic proof for first-order logic with inductive definitions, Journal of Logic and Computation 35 (2025).

[16] S. Berardi, M. Tatsuta, Explicit induction is not equivalent to cyclic proofs for classical logic with inductive definitions, Logical methods in computer science 15 (2019).

[17] M. Gelfond, V. Lifschitz, The stable model semantics for logic programming, in: R. A. Kowalski, K. A. Bowen (Eds.), ICLP/SLP, MIT Press, 1988, pp. 1070–1080.

[18] P. Hou, J. Wittocx, M. Denecker, A deductive system for PC(ID), in: C. Baral, G. Brewka, J. S. Schlipf (Eds.), LPNMR, volume 4483 of *Lecture Notes in Computer Science*, Springer, 2007, pp. 162–174.

[19] P. Hou, M. Denecker, A deductive system for FO(ID) based on least fixpoint logic, in: E. Erdem, F. Lin, T. Schaub (Eds.), LPNMR, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 129–141.

[20] J. Brotherston, N. Gorogiannis, R. L. Petersen, A generic cyclic theorem prover, in: R. Jhala, A. Igarashi (Eds.), Programming Languages and Systems, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 350–367.

[21] M. E. Szabó (Ed.), The collected papers of Gerhard Gentzen., Studies in logic and the foundations of mathematics, North-Holland, Amsterdam, 1969.