

# A Cut-free, Sound and Complete Russellian Theory of Definite Descriptions<sup>\*</sup>

Andrzej Indrzejczak<sup>1</sup>[0000–0003–4063–1651] and Nils Kürbis<sup>1</sup>[0000–0002–3651–545]

Department of Logic, University of Lodz, Poland  
andrzej.indrzejczak@filhist.uni.lodz.pl  
nils.kurbis@filhist.uni.lodz.pl

**Abstract.** We present a sequent calculus for first-order logic with lambda terms and definite descriptions. The theory formalised by this calculus is essentially Russellian, but avoids some of its well known drawbacks and treats definite description as genuine terms. A constructive proof of the cut elimination theorem and a Henkin-style proof of completeness are the main results of this contribution.

**Keywords:** Definite Descriptions · Predicate abstracts · Sequent Calculus · Cut Elimination

## 1 Introduction

Definite descriptions (DD) are complex terms commonly applied not only in natural languages but also in mathematics and computer science. In formal languages they are usually expressed by means of the iota operator, which forms terms from formulas. Thus  $\iota x\varphi$  means ‘the (only)  $x$  satisfying  $\varphi$ ’. A DD aims to denote a unique object by virtue of a property that only it has. Sometimes a DD fails, because nothing or more than one thing has the property. A DD that succeeds to denote only one object is *proper*; otherwise it is *improper*.

Definite descriptions, proper and improper, are ubiquitous not only in natural languages but also in mathematics and science (like the proper ‘the sum of 7 and 5’ or the improper ‘the square root of  $n$ ’). In formal languages the application of functional terms is the prevailing way of representing complex names. However, applying DD can outrun functional terms in many ways, since they are more expressive than functional terms, in the sense that an arbitrary functional term  $f^n(t_1, \dots, t_n)$  can be represented as a description  $\iota xF^{n+1}(x, t_1, \dots, t_n)$ , where  $F$  is a predicate corresponding to the function  $f$ . On the other hand, not every definite description, even if proper, can be expressed using functional terms; it is possible only in the case of predicates expressing functional relations, whereas every sentence can be used to form a DD. For example, both ‘the father of Ben’

---

<sup>\*</sup> Funded by the European Union (ERC, ExtenDD, project number: 101054714). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

and ‘the daughter of Mary’ may be represented as terms using the iota operator, but only the first may be represented as a functional term. Moreover, even if we can use functional terms instead of DD we enrich a language with another sort of functors in addition to predicates. This has an impact on the formalisation of valid arguments in which very often the conclusion follows on the basis of the content expressed by functional terms which is directly expressed by predicates. For example: ‘Adam has children’ follows from ‘Adam is the father of Ben’. However to prove its validity, its formal representation  $a = f(b) \vdash \exists x Cxa$  requires two enthymematic premisses:  $\forall xy(Mxy \vee Fxy \leftrightarrow Cyx)$  and  $\forall xy(x = f(y) \leftrightarrow Fxy)$ . Let us call the latter premiss a bridge principle allowing us to transfer information conveyed by predicates to related functions and vice versa. In general they have a form:  $\forall x_1, \dots, x_n, y(y = f^n(x_1, \dots, x_n) \leftrightarrow F^{n+1}(y, x_1, \dots, x_n))$  and show how the information encoded by the functional predicates is represented by predicates. In the case of using DD instead of functional terms we do not need such extra bridge principles, whereas in languages with functional terms they are necessary in an analysis of obviously valid arguments.<sup>1</sup>

The usefulness of formal devices like the iota operator and other term-forming operators has recently been better recognised (cf. Tennant’s [32] or Scott and Benzmüller’s implementation of free logic using proof assistant *Isabelle/HOL* [3]) also in the fields connected with computer science, like differential dynamic logic used for verification of hybrid systems [5] or description logics (see [1] or [25]). Logics with DD are often implemented to enable formalisation of deep philosophical problems. e.g. Anselm’s ontological argument (see the work by Oppenheimer and Zalta using the automated reasoning tool *PROVER9* [26] or its encoding by Blunson [4]).

Since several rival theories of DD were formulated, the applicability and potential usefulness of DD was underestimated so far. It leads to a question which approach is the best one, at least for some specific kind of applications. In this paper we focus on the Russellian approach to definite descriptions ([28] and [35]) which plays a central role in this area. Although Russell’s theory of DD has some controversial points, it became a standard point of reference of almost all works devoted to the analysis of definite descriptions. Moreover, it is still widely accepted by formal logicians as a proper way of handling descriptions; the scores of textbooks that use it as their official theory of definite descriptions count as witnesses for this claim. Russell’s theory has also strong affinities to logics closely connected with applications in constructive mathematics and computer science like the logic of the existence predicate by Scott [30] or the definedness logic (or the logic of partial terms) of Beeson [2] and Feferman [8]. These connections were elaborated in [14].

Russell treated DD as incomplete signs and defined their use by contextual definitions of the form:

$$\psi[x/iy\varphi] := \exists x(\forall y(\varphi \leftrightarrow y = x) \wedge \psi)$$

<sup>1</sup> Some other advantages of using DD instead of functional terms are discussed in more detail in [17].

but this solution leads to scoping difficulties if  $\psi$  is not elementary.  $\neg\psi[x/iy\varphi]$ , e.g., is ambiguous: is the whole formula negated or only the predicate  $\psi$ ? The method which Russell introduced in [35] to draw scope distinctions is rather clumsy. Fortunately, it is possible to develop a logic which treats DD as genuine terms and yet retains desirable features of the Russellian approach. Such a logic was formalised as a natural deduction system by Kalish, Montague, and Mar [18] and by Francez and Więckowski [11]. These systems involve complex rules and axioms, but recently Indrzejczak [16] provided an analytic and cut-free sequent calculus equivalent to the Russellian logic as formalised in [18]. However, in all these systems the formal counterpart of the Russellian policy of eliminating DD from sentences must be restricted to predicate letters, which is connected with the scoping difficulties of the Russellian approach just mentioned.

Can we offer any improvement on the state of the art? A possible strategy of avoiding these problems is to treat DD by means of a binary quantifier; this approach was formally developed by Kürbis (cf. [19], [20], [21], [22], [23]). However, if we want to treat DD as terms, then the introduction of the lambda operator to construct complex predicate abstracts from formulas offers a good solution.  $\lambda x\varphi$  means ‘the property of being  $\varphi$ ’ and applied to some term, in particular to a DD, forms a formula called a lambda atom. This device was introduced into studies of modal predicate logic by Thomason and Stalnaker [31], and the idea was further developed by Bressan [6] and Fitting [9], in particular, to distinguish between *de dicto* and *de re* reading of modal operators. Independently, this technique was used by Scales [29] in his formulation of attributional logic, where Aristotle’s distinction between the negation of a sentence and of a predicate is formally expressible. In fact, Scales seems to be the first one to apply predicate abstraction to formalise a theory of DD which relates closely to Russell’s. Predicate abstracts were also successfully applied by Fitting and Mendelsohn [10] to obtain a theory of DD in a modal setting. This approach, with slight modifications, was further developed independently by Orlandelli [27] and Indrzejczak [12] to obtain cut-free sequent calculi for modal logics with DD and predicate abstracts.

In this article we focus on a different logic **RL**, first introduced in [17], which also combines the iota and lambda operators. It avoids the shortcomings of the Russellian approach while saving all its plausible features. Predicate abstracts permit us to draw scope distinctions rather more elegantly than with the Russellian scope markers and their application is more general. **RL** is essentially Russellian but with DD treated as genuine terms. Nonetheless, the reductionist aspect of Russell’s approach is retained in several ways. On the level of syntax the occurrences of DD are restricted to arguments of predicate abstracts to form lambda atoms. On the level of semantics DD are not defined by an interpretation function but by satisfaction clauses for lambda atoms. Eventually, on the level of calculus DD cannot be instantiated for variables in quantifier rules but are subject to special rules for lambda atoms. This strict connection of DD with predicate abstracts avoids disadvantages of the Russellian approach connected

with scoping difficulties, and, at the same time, simplifies proofs of metalogical properties.

**RL** was originally characterised semantically and formalised as an analytic tableau calculus in [17], where it was also applied for proving the Craig interpolation theorem. Here we are completing the research on **RL** by providing an adequate sequent calculus for which the cut elimination theorem is proved constructively. We characterise the language, semantics and axiomatisation of **RL** in section 2. Then we present the sequent calculus GRL for **RL** and show its equivalence with an axiomatic Hilbert style system HRL. Section 4 contains a proof of the cut elimination theorem, and section 5 a Henkin-style proof of completeness. The paper finishes with some comparative remarks.

## 2 Preliminaries

The language  $\mathcal{L}$  of **RL** is standard, except that it contains the operators  $\iota$  and  $\lambda$ . Following the remarks on the functional terms from the Introduction, as well as the original Russellian attitude towards terms, the ‘official’ language has neither constant nor function symbols; in the completeness proof we add constants solely for the purpose of constructing models from consistent sets. As is customary in proof theoretic investigations since Gentzen, we distinguish free and bound variables graphically in deductions. It is not customary to make this distinction in semantics, and so there we won’t make it either. This blend of two customs should not lead to confusion, and we are following Fitting and Mendelsohn [10] in this respect. There are two disjoint sets  $VAR$  of variables and  $PAR$  of parameters. The former plays the role of the bound, the latter of the free variables in the presentation of the proof theory of **RL**; in the presentation of the semantics, this restriction is relaxed and members of  $VAR$  are permitted as free variables. The *terms* of the language in the strict sense are the variables and parameters. Expressions formed by  $\iota$  are admitted as terms in a more general sense: their application is restricted to predicate abstracts and they are called quasi-terms. We mention only the following formation rules for the more general notion of a formula used in the semantics:

- If  $P^n$  is a predicate symbol (including  $=$ ) and  $t_1 \dots t_n \in VAR \cup PAR$ , then  $P^n(t_1, \dots, t_n)$  is a formula (atomic formula).
- If  $\varphi$  is a formula, then  $(\lambda x\varphi)$  is a predicate abstract.
- If  $\varphi$  is a formula, then  $\iota x\varphi$  is a quasi-term.
- If  $\varphi$  is a predicate abstract and  $t$  a term or quasi-term, then  $\varphi t$  is a formula (lambda atom).

$\varphi[x/t]$  denotes the result of replacing  $x$  by  $t$  in  $\varphi$ . To save space, we’ll often write  $\varphi_t^x$  instead of  $\varphi[x/t]$ . If  $t$  is a variable  $y$ , it is assumed that  $y$  is free for  $x$  in  $\varphi$ , that is, no occurrence of  $y$  becomes bound in  $\varphi$  in the replacement. To save space and simplify things in the statement of semantics and in the completeness proof in section 4, we treat  $\forall, \rightarrow, \exists$  as defined notions.

A *model* is a structure  $M = \langle D, I \rangle$ , where for each  $n$ -argument predicate  $P^n$ ,  $I(P^n) \subseteq D^n$ . An *assignment*  $v$  is a function  $v : VAR \cup PAR \rightarrow D$ . An  $x$ -*variant*  $v'$  of  $v$  agrees with  $v$  on all arguments, save possibly  $x$ . We write  $v_o^x$  to denote the  $x$ -variant of  $v$  with  $v_o^x(x) = o$ . The notion of *satisfaction* of a formula  $\varphi$  with  $v$ , in symbols  $M, v \models \varphi$ , is defined as follows, where  $t \in VAR \cup PAR$ :

$$\begin{aligned}
M, v \models P^n(t_1, \dots, t_n) & \text{ iff } \langle v(t_1), \dots, v(t_n) \rangle \in I(P^n) \\
M, v \models t_1 = t_2 & \text{ iff } v(t_1) = v(t_2) \\
M, v \models (\lambda x \psi)t & \text{ iff } M, v_o^x \models \psi, \text{ where } o = v(t) \\
M, v \models (\lambda x \psi) \iota y \varphi & \text{ iff there is an } o \in D \text{ such that } M, v_o^x \models \psi, \text{ and} \\
& \quad M, v_o^x \models \varphi[y/x], \text{ and for any } y\text{-variant } v' \text{ of } v_o^x, \\
& \quad \text{if } M, v' \models \varphi, \text{ then } v'(y) = o \\
M, v \models \neg \varphi & \text{ iff } M, v \not\models \varphi, \\
M, v \models \varphi \wedge \psi & \text{ iff } M, v \models \varphi \text{ and } M, v \models \psi, \\
M, v \models \forall x \varphi & \text{ iff } M, v_o^x \models \varphi, \text{ for all } o \in D.
\end{aligned}$$

A formula  $\varphi$  is *satisfiable* if there are a model  $M$  and an assignment  $v$  such that  $M, v \models \varphi$ . A formula is *valid* if, for all models  $M$  and assignments  $v$ ,  $M, v \models \varphi$ . Semantically, HRL is identified with the set of valid formulas, **RL** with the set of valid sequents. A set of formulas  $\Gamma$  is *satisfiable* iff there is some structure  $M$  and an assignment  $v$  such that  $M$  satisfies every member of  $\Gamma$  with  $v$ . A sequent  $\Gamma \Rightarrow \Delta$  is satisfied by a structure  $M$  with an assignment  $v$  if and only if, if for all  $\varphi \in \Gamma$ ,  $M, v \models \varphi$ , then for some  $\psi \in \Delta$ ,  $M, v \models \psi$ . We symbolise this by  $M, v \models \Gamma \Rightarrow \Delta$ . A sequent  $\Gamma \Rightarrow \Delta$  is *valid* iff it is satisfied by every structure with every assignment  $v$ . In this case we write  $\models \Gamma \Rightarrow \Delta$ .

Note that we do not characterise DD semantically by means of interpretation function  $I$  as it is usually done (for example in [10], [27]). The syntactic restriction making DD only arguments in lambda atoms allows us to define them together as a separate satisfaction clause instead. It is closer to the original Russellian treatment of descriptions and simplifies the completeness proof.

Before presenting the sequent calculus, we briefly give the Hilbert system HRL. As we noted Russell treated DD as incomplete symbols and eliminated them by means of contextual definitions. Adopting the following axiom corresponding to his definitions would be too simplistic:

$$R \quad \psi(\iota y \varphi) \leftrightarrow \exists x(\forall y(\varphi \leftrightarrow y = x) \wedge \psi)$$

$R$  must be restricted to atomic  $\psi$  or it is necessary to add means for marking scope distinctions. Whitehead and Russell chose the latter part, but their method is far from ideal. It is possible to avoid the problem in more elegant fashion with the help of a  $\lambda$  operator. In particular, we can use it to distinguish the application of the negated predicate  $\neg\psi$  to  $\iota y \varphi$  from negating the application of  $\psi$  to it. In the present context scoping difficulties arise only in relation to DD, and the problem is solved by restricting predication on DD to predicate abstracts. Accordingly, atomic formulas are built from predicate symbols and variables/parameters only. This is in full accordance with Russell, since the language of *Principia* contains no primitive constant and function symbols: they are introduced by contextual

definitions by means of DD. We modify  $R$  to reflect the restriction that  $\iota$  terms require  $\lambda$  abstracts:

$$R_\lambda \quad (\lambda x\psi)\iota y\varphi \leftrightarrow \exists x(\forall y(\varphi \leftrightarrow y = x) \wedge \psi)$$

This way we avoid problems with scope while permitting complex as well as primitive predicates to be applied to DD. The axiomatic system HRL for our logic **RL** results from a standard axiomatization of pure first-order logic with identity and quantifier rules restricted to parameters by adding the axiom  $R_\lambda$  and  $\beta$ -conversion for  $\lambda$  but restricted again to parameters:  $(\lambda x\psi)t \leftrightarrow \psi[x/t]$ , where  $t$  is a parameter. The adequacy of HRL will be demonstrated below.

### 3 Sequent Calculus

We now formalise the Russellian logic **RL** as a sequent calculus GRL. Sequents  $\Gamma \Rightarrow \Delta$  are ordered pairs of finite multisets of formulas, called the antecedent and the succedent, respectively. GRL is essentially the calculus G1c of Troelstra and Schwichtenberg [34] with rules for identity and lambda atoms: see Fig. 1.

Let us recall that formulas displayed in the schemata are active, whereas the remaining ones are parametric, or form a context. In particular, all active formulas in the premisses are called side formulas, and the one in the conclusion is the principal formula of the respective rule application. Proofs are defined in the standard way as finite trees with nodes labelled by sequents. The height of a proof  $\mathcal{D}$  of  $\Gamma \Rightarrow \Delta$  is defined as the number of nodes of the longest branch in  $\mathcal{D}$ .  $\vdash_k \Gamma \Rightarrow \Delta$  means that  $\Gamma \Rightarrow \Delta$  has a proof with height at most  $k$ .  $\vdash$  means that there is a proof of the expression standing to its right, be it a formula (in the case of HRL) or a sequent (in the case of GRL).

We need some auxiliary results. In particular, since  $(= -)$  is Leibniz' Principle restricted to atomic formulas, we must prove its unrestricted form.

**Lemma 1.** 1.  $\vdash b_1 = b_2, \varphi[x/b_1] \Rightarrow \varphi[x/b_2]$ , for any formula  $\varphi$ .  
2. If  $\vdash_k \Gamma \Rightarrow \Delta$ , then  $\vdash_k \Gamma[b_1/b_2] \Rightarrow \Delta[b_1/b_2]$ , where  $k$  is the height of a proof.

*Proof.* 1. follows by induction over the complexity of formulas, which is standard for all cases except those concerning lambda atoms with DD. We note that  $\varphi_{b_c}^{z_y}$  is the same as  $\varphi_{c_b}^{y_z}$ , etc. We write  $[(\lambda x\psi)\iota y\varphi]_{b_1}^z$  to denote substitutions in lambda atoms in more readable fashion. To simplify proofs applications of weakening and contraction rules to derive shared contexts are omitted from now on. Let  $\mathcal{D}$  be the following deduction, where the leaves are axioms and  $c$  a fresh parameter:

$$(\iota_2 \Rightarrow) \frac{\varphi_{c b_1}^{y_z} \Rightarrow \varphi_{c b_1}^{y_z} \quad \varphi_{a b_1}^{y_z} \Rightarrow \varphi_{a b_1}^{y_z} \quad c = a \Rightarrow c = a}{[(\lambda x\psi)\iota y\varphi]_{b_1}^z, \varphi_{a b_1}^{y_z}, \varphi_{c b_1}^{y_z} \Rightarrow c = a}$$

Then we derive  $\vdash b_1 = b_2, [(\lambda x\psi)\iota y\varphi]_{b_1}^z \Rightarrow [(\lambda x\psi)\iota y\varphi]_{b_2}^z$ :

$$\begin{array}{l}
(Cut) \frac{\Gamma \Rightarrow \Delta, \varphi \quad \varphi, \Pi \Rightarrow \Sigma}{\Gamma, \Pi \Rightarrow \Delta, \Sigma} \quad (AX) \varphi \Rightarrow \varphi \\
(W \Rightarrow) \frac{\Gamma \Rightarrow \Delta}{\varphi, \Gamma \Rightarrow \Delta} \quad (\Rightarrow W) \frac{\Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, \varphi} \\
(C \Rightarrow) \frac{\varphi, \varphi, \Gamma \Rightarrow \Delta}{\varphi, \Gamma \Rightarrow \Delta} \quad (\Rightarrow C) \frac{\Gamma \Rightarrow \Delta, \varphi, \varphi}{\Gamma \Rightarrow \Delta, \varphi} \\
(\neg \Rightarrow) \frac{\Gamma \Rightarrow \Delta, \varphi}{\neg \varphi, \Gamma \Rightarrow \Delta} \quad (\Rightarrow \neg) \frac{\varphi, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, \neg \varphi} \\
(\Rightarrow \wedge) \frac{\Gamma \Rightarrow \Delta, \varphi \quad \Gamma \Rightarrow \Delta, \psi}{\Gamma \Rightarrow \Delta, \varphi \wedge \psi} \quad (\wedge \Rightarrow) \frac{\varphi, \psi, \Gamma \Rightarrow \Delta}{\varphi \wedge \psi, \Gamma \Rightarrow \Delta} \\
(\Rightarrow \vee) \frac{\varphi, \Gamma \Rightarrow \Delta \quad \psi, \Gamma \Rightarrow \Delta}{\varphi \vee \psi, \Gamma \Rightarrow \Delta} \quad (\Rightarrow \vee) \frac{\Gamma \Rightarrow \Delta, \varphi, \psi}{\Gamma \Rightarrow \Delta, \varphi \vee \psi} \\
(\Rightarrow \rightarrow) \frac{\Gamma \Rightarrow \Delta, \varphi \quad \psi, \Gamma \Rightarrow \Delta}{\varphi \rightarrow \psi, \Gamma \Rightarrow \Delta} \quad (\Rightarrow \rightarrow) \frac{\varphi, \Gamma \Rightarrow \Delta, \psi}{\Gamma \Rightarrow \Delta, \varphi \rightarrow \psi} \\
(\leftrightarrow \Rightarrow) \frac{\Gamma \Rightarrow \Delta, \varphi, \psi \quad \varphi, \psi, \Gamma \Rightarrow \Delta}{\varphi \leftrightarrow \psi, \Gamma \Rightarrow \Delta} \quad (\forall \Rightarrow) \frac{\varphi[x/b], \Gamma \Rightarrow \Delta}{\forall x \varphi, \Gamma \Rightarrow \Delta} \\
(\Rightarrow \leftrightarrow) \frac{\varphi, \Gamma \Rightarrow \Delta, \psi \quad \psi, \Gamma \Rightarrow \Delta, \varphi}{\Gamma \Rightarrow \Delta, \varphi \leftrightarrow \psi} \quad (\Rightarrow \forall) \frac{\Gamma \Rightarrow \Delta, \varphi[x/a]}{\Gamma \Rightarrow \Delta, \forall x \varphi} \\
(\exists \Rightarrow) \frac{\varphi[x/a], \Gamma \Rightarrow \Delta}{\exists x \varphi, \Gamma \Rightarrow \Delta} \quad (\Rightarrow \exists) \frac{\Gamma \Rightarrow \Delta, \varphi[x/b]}{\Gamma \Rightarrow \Delta, \exists x \varphi} \\
(= -) \frac{\varphi[x/b_2], \Gamma \Rightarrow \Delta}{b_1 = b_2, \varphi[x/b_1], \Gamma \Rightarrow \Delta} \quad (= +) \frac{b = b, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta} \\
(\lambda \Rightarrow) \frac{\psi[x/b], \Gamma \Rightarrow \Delta}{(\lambda x \psi)b, \Gamma \Rightarrow \Delta} \quad (\Rightarrow \lambda) \frac{\Gamma \Rightarrow \Delta, \psi[x/b]}{\Gamma \Rightarrow \Delta, (\lambda x \psi)b} \\
(\iota_1 \Rightarrow) \frac{\varphi[y/a], \psi[x/a], \Gamma \Rightarrow \Delta}{(\lambda x \psi) \iota y \varphi, \Gamma \Rightarrow \Delta} \\
(\iota_2 \Rightarrow) \frac{\Gamma \Rightarrow \Delta, \varphi[y/b_1] \quad \Gamma \Rightarrow \Delta, \varphi[y/b_2] \quad b_1 = b_2, \Gamma \Rightarrow \Delta}{(\lambda x \psi) \iota y \varphi, \Gamma \Rightarrow \Delta} \\
(\Rightarrow \iota) \frac{\Gamma \Rightarrow \Delta, \varphi[y/b] \quad \Gamma \Rightarrow \Delta, \psi[x/b] \quad \varphi[y/a], \Gamma \Rightarrow \Delta, a = b}{\Gamma \Rightarrow \Delta, (\lambda x \psi) \iota y \varphi}
\end{array}$$

where  $a$  is a fresh parameter (Eigenvariable), not present in  $\Gamma, \Delta$  and  $\varphi$ , whereas  $b, b_1, b_2$  are arbitrary parameters.  $\varphi$  in  $(= -)$  is an atomic formula.

**Fig. 1.** Calculus GRL

$$\begin{array}{l}
(\Rightarrow \iota) \frac{b_1 = b_2, \varphi_{a b_1}^y \Rightarrow \varphi_{a b_2}^y \quad b_1 = b_2, \psi_{a b_1}^x \Rightarrow \psi_{a b_2}^x \quad \mathcal{D}}{b_1 = b_2, \varphi_{a b_1}^y, \psi_{a b_1}^x, [(\lambda x \psi) \iota y \varphi]_{b_1}^z \Rightarrow [(\lambda x \psi) \iota y \varphi]_{b_2}^z} \\
(\iota_1 \Rightarrow) \frac{b_1 = b_2, [(\lambda x \psi) \iota y \varphi]_{b_1}^z, [(\lambda x \psi) \iota y \varphi]_{b_1}^z \Rightarrow [(\lambda x \psi) \iota y \varphi]_{b_2}^z}{b_1 = b_2, [(\lambda x \psi) \iota y \varphi]_{b_1}^z \Rightarrow [(\lambda x \psi) \iota y \varphi]_{b_2}^z} \\
(C \Rightarrow) \frac{b_1 = b_2, [(\lambda x \psi) \iota y \varphi]_{b_1}^z \Rightarrow [(\lambda x \psi) \iota y \varphi]_{b_2}^z}{b_1 = b_2, [(\lambda x \psi) \iota y \varphi]_{b_1}^z \Rightarrow [(\lambda x \psi) \iota y \varphi]_{b_2}^z}
\end{array}$$

The two left leaves are provable by the induction hypothesis (if  $b_1, b_2$  are not present in  $\psi$  or  $\varphi$ , we have an axiomatic sequent).

The proof of 2 is by a standard induction on the height of proofs; the rules for lambda atoms with DD are treated similarly to the rules for quantifiers.  $\square$

Let us now show that the Russellian axiom  $R_\lambda$  is provable in GRL. We will provide proofs for two sequents corresponding to two implications. Let  $\mathcal{D}$  be:

$$(\iota_2 \Rightarrow) \frac{\varphi_a^y \Rightarrow \varphi_a^y \quad \varphi_{a_1}^y \Rightarrow \varphi_{a_1}^y \quad a_1 = a \Rightarrow a_1 = a}{(\lambda x \psi) \iota y \varphi, \varphi_a^y, \varphi_{a_1}^y \Rightarrow a_1 = a}$$

The following establishes one half of  $R_\lambda$ :

$$\begin{array}{l}
(\Rightarrow \leftrightarrow) \frac{\mathcal{D} \quad \varphi_a^y, a_1 = a \Rightarrow \varphi_{a_1}^y}{(\lambda x \psi) \iota y \varphi, \varphi_a^y \Rightarrow \varphi_{a_1}^y \leftrightarrow a_1 = a} \\
(\Rightarrow \forall) \frac{(\lambda x \psi) \iota y \varphi, \varphi_a^y \Rightarrow \varphi_{a_1}^y \leftrightarrow a_1 = a}{(\lambda x \psi) \iota y \varphi, \varphi_a^y \Rightarrow \forall y (\varphi \leftrightarrow y = a)} \quad \psi_a^x \Rightarrow \psi_a^x \\
(\Rightarrow \wedge) \frac{(\lambda x \psi) \iota y \varphi, \psi_a^x, \varphi_a^y \Rightarrow \forall y (\varphi \leftrightarrow y = a) \quad \psi_a^x \Rightarrow \psi_a^x}{(\lambda x \psi) \iota y \varphi, \psi_a^x, \varphi_a^y \Rightarrow \forall y (\varphi \leftrightarrow y = a) \wedge \psi_a^x} \\
(\Rightarrow \exists) \frac{(\lambda x \psi) \iota y \varphi, \psi_a^x, \varphi_a^y \Rightarrow \exists x (\forall y (\varphi \leftrightarrow y = x) \wedge \psi)}{(\lambda x \psi) \iota y \varphi, \psi_a^x, \varphi_a^y \Rightarrow \exists x (\forall y (\varphi \leftrightarrow y = x) \wedge \psi)} \\
(\iota_1 \Rightarrow) \frac{(\lambda x \psi) \iota y \varphi, (\lambda x \psi) \iota y \varphi \Rightarrow \exists x (\forall y (\varphi \leftrightarrow y = x) \wedge \psi)}{(\lambda x \psi) \iota y \varphi, (\lambda x \psi) \iota y \varphi \Rightarrow \exists x (\forall y (\varphi \leftrightarrow y = x) \wedge \psi)} \\
(C \Rightarrow) \frac{(\lambda x \psi) \iota y \varphi, (\lambda x \psi) \iota y \varphi \Rightarrow \exists x (\forall y (\varphi \leftrightarrow y = x) \wedge \psi)}{(\lambda x \psi) \iota y \varphi \Rightarrow \exists x (\forall y (\varphi \leftrightarrow y = x) \wedge \psi)}
\end{array}$$

where the only nonaxiomatic sequent is provable by lemma 1.1. Next, where  $\mathcal{D}$  is:

$$\begin{array}{l}
(\leftrightarrow \Rightarrow) \frac{\varphi_b^y \Rightarrow \varphi_b^y \quad b = a \Rightarrow b = a}{\varphi_b^y \leftrightarrow b = a, \varphi_b^y \Rightarrow b = a} \\
(\forall \Rightarrow) \frac{\varphi_b^y \leftrightarrow b = a, \varphi_b^y \Rightarrow b = a}{\forall y (\varphi \leftrightarrow y = a), \varphi_b^y \Rightarrow b = a}
\end{array}$$

the following establishes the other half of  $R_\lambda$ :

$$\begin{array}{l}
(= +) \frac{a = a \Rightarrow a = a}{\Rightarrow a = a} \quad \varphi_a^y \Rightarrow \varphi_a^y \\
(\leftrightarrow \Rightarrow) \frac{\varphi_a^y \leftrightarrow a = a \Rightarrow \varphi_a^y}{\forall y (\varphi \leftrightarrow y = a) \Rightarrow \varphi_a^y} \\
(\forall \Rightarrow) \frac{\varphi_a^y \leftrightarrow a = a \Rightarrow \varphi_a^y}{\forall y (\varphi \leftrightarrow y = a) \Rightarrow \varphi_a^y} \\
(\Rightarrow \iota) \frac{\psi_a^x \Rightarrow \psi_a^x \quad \forall y (\varphi \leftrightarrow y = a), \psi_a^x \Rightarrow (\lambda x \psi) \iota y \varphi}{(\wedge \Rightarrow) \frac{\forall y (\varphi \leftrightarrow y = a), \psi_a^x \Rightarrow (\lambda x \psi) \iota y \varphi}{\forall y (\varphi \leftrightarrow y = a) \wedge \psi_a^x \Rightarrow (\lambda x \psi) \iota y \varphi}} \quad \mathcal{D} \\
(\exists \Rightarrow) \frac{\exists x (\forall y (\varphi \leftrightarrow y = x) \wedge \psi) \Rightarrow (\lambda x \psi) \iota y \varphi}{\exists x (\forall y (\varphi \leftrightarrow y = x) \wedge \psi) \Rightarrow (\lambda x \psi) \iota y \varphi}
\end{array}$$

Conversely, the three rules for lambda atoms with DD are derivable in G1 with  $R_\lambda$  added in the form of two axiomatic sequents. To derive  $(\iota_1 \Rightarrow)$ , let  $R_\lambda^{\Rightarrow}$  be  $(\lambda x \psi) \iota y \varphi \Rightarrow \exists x (\forall y (\varphi \leftrightarrow y = x) \wedge \psi)$ :



$$\begin{array}{c}
 (= +) \frac{a = a \Rightarrow a = a}{\Rightarrow a = a} \quad \varphi_a^y, \psi_a^x, \Gamma \Rightarrow \Delta \\
 (\leftrightarrow \Rightarrow) \frac{\varphi_a^y \leftrightarrow a = a, \psi_a^x, \Gamma \Rightarrow \Delta}{\forall y(\varphi \leftrightarrow y = a), \psi_a^x, \Gamma \Rightarrow \Delta} \\
 (\forall \Rightarrow) \frac{\forall y(\varphi \leftrightarrow y = a), \psi_a^x, \Gamma \Rightarrow \Delta}{\forall y(\varphi \leftrightarrow y = a) \wedge \psi_a^x, \Gamma \Rightarrow \Delta} \\
 (\wedge \Rightarrow) \frac{\forall y(\varphi \leftrightarrow y = a) \wedge \psi_a^x, \Gamma \Rightarrow \Delta}{\exists x(\forall y(\varphi \leftrightarrow y = x) \wedge \psi), \Gamma \Rightarrow \Delta} \\
 (\exists \Rightarrow) \frac{\exists x(\forall y(\varphi \leftrightarrow y = x) \wedge \psi), \Gamma \Rightarrow \Delta}{(\lambda x \psi) \iota y \varphi, \Gamma \Rightarrow \Delta} \\
 (Cut) \frac{R_{\lambda}^{\Rightarrow}}{\quad}
 \end{array}$$

To derive  $(\iota_2 \Rightarrow)$ , use  $(Cut)$  with  $(\lambda x \psi) \iota y \varphi \Rightarrow \exists x(\forall y(\varphi \leftrightarrow y = x) \wedge \psi)$  and:

$$\begin{array}{c}
 \Gamma \Rightarrow \Delta, \varphi_{b_1}^y \quad (\leftrightarrow \Rightarrow) \frac{\Gamma \Rightarrow \Delta, \varphi_{b_2}^y \quad (= -) \frac{b_1 = b_2, \Gamma \Rightarrow \Delta}{b_1 = a, b_2 = a, \Gamma \Rightarrow \Delta}}{b_1 = a, \varphi_{b_2}^y \leftrightarrow b_2 = a, \Gamma \Rightarrow \Delta} \\
 (\leftrightarrow \Rightarrow) \frac{\Gamma \Rightarrow \Delta, \varphi_{b_1}^y}{\varphi_{b_1}^y \leftrightarrow b_1 = a, \varphi_{b_2}^y \leftrightarrow b_2 = a, \Gamma \Rightarrow \Delta} \\
 (\forall \Rightarrow) \frac{\varphi_{b_1}^y \leftrightarrow b_1 = a, \varphi_{b_2}^y \leftrightarrow b_2 = a, \Gamma \Rightarrow \Delta}{\forall y(\varphi \leftrightarrow y = a), \forall y(\varphi \leftrightarrow y = a), \Gamma \Rightarrow \Delta} \\
 (C \Rightarrow) \frac{\forall y(\varphi \leftrightarrow y = a), \forall y(\varphi \leftrightarrow y = a), \Gamma \Rightarrow \Delta}{\forall y(\varphi \leftrightarrow y = a), \psi_a^x, \Gamma \Rightarrow \Delta} \\
 (\wedge \Rightarrow) \frac{\forall y(\varphi \leftrightarrow y = a), \psi_a^x, \Gamma \Rightarrow \Delta}{\forall y(\varphi \leftrightarrow y = a) \wedge \psi_a^x, \Gamma \Rightarrow \Delta} \\
 (\exists \Rightarrow) \frac{\forall y(\varphi \leftrightarrow y = a) \wedge \psi_a^x, \Gamma \Rightarrow \Delta}{\exists x(\forall y(\varphi \leftrightarrow y = x) \wedge \psi), \Gamma \Rightarrow \Delta}
 \end{array}$$

The following derives  $(\Rightarrow \iota)$ :

$$\begin{array}{c}
 \varphi_a^y, \Gamma \Rightarrow \Delta, a = b \quad (Cut) \frac{\Gamma \Rightarrow \Delta, \varphi_b^y \quad a = b, \varphi_b^y \Rightarrow \varphi_a^y}{a = b, \Gamma \Rightarrow \Delta, \varphi_a^y} \\
 (\Rightarrow \leftrightarrow) \frac{\varphi_a^y, \Gamma \Rightarrow \Delta, a = b}{\Gamma \Rightarrow \Delta, \varphi_a^y \leftrightarrow a = b} \\
 (\Rightarrow \forall) \frac{\Gamma \Rightarrow \Delta, \varphi_a^y \leftrightarrow a = b}{\Gamma \Rightarrow \Delta, \forall y(\varphi \leftrightarrow y = b)} \\
 (\Rightarrow \wedge) \frac{\Gamma \Rightarrow \Delta, \forall y(\varphi \leftrightarrow y = b) \quad \Gamma \Rightarrow \Delta, \psi_b^x}{\Gamma \Rightarrow \Delta, \forall y(\varphi \leftrightarrow y = b) \wedge \psi_b^x} \\
 (\Rightarrow \exists) \frac{\Gamma \Rightarrow \Delta, \forall y(\varphi \leftrightarrow y = b) \wedge \psi_b^x}{\Gamma \Rightarrow \Delta, \exists x(\forall y(\varphi \leftrightarrow y = x) \wedge \psi)}
 \end{array}$$

where the right premiss of  $(Cut)$  is provable by lemma 1.1, and the conclusion of the rule follows by  $(Cut)$  with  $\exists x(\forall y(\varphi \leftrightarrow y = x) \wedge \psi) \Rightarrow (\lambda x \psi) \iota y \varphi$ .

Since the proofs of the interderivability of the axiom of  $\lambda$  conversion and  $(\lambda \Rightarrow)$ ,  $(\Rightarrow \lambda)$  are trivial we are done and conclude with:

**Theorem 1.**  $\vdash_{HRL} \varphi$  iff  $\vdash_{GRL} \Rightarrow \varphi$

## 4 Cut Elimination

We will show that  $(Cut)$  is eliminable from every proof in GRL using the general strategy of cut elimination proofs applied originally for hypersequent calculi in Metcalfe, Olivetti and Gabbay [24], which works well also in the context of standard sequent calculi (see [15]). Such a proof has a particularly simple structure and allows us to avoid many complexities inherent in other methods of proving cut elimination. In particular, we avoid well known problems with contraction, since two auxiliary lemmata deal with this problem in advance. We assume that all proofs are regular in the sense that every parameter  $a$  which is

fresh by the side condition of the respective rule must be fresh in the entire proof, not only on the branch where the application of this rule takes place. There is no loss of generality since every proof may be systematically transformed into a regular one by lemma 1.2. The following notions are crucial for the proof:

1. The cut-degree is the complexity of the cut-formula  $\varphi$ , i.e. the number of logical constants (connectives, quantifiers and operators) occurring in  $\varphi$ ; it is denoted by  $d\varphi$ .
2. The proof-degree ( $d\mathcal{D}$ ) is the maximal cut-degree in  $\mathcal{D}$ .

The proof of the cut elimination theorem is based on two lemmata which successively make a reduction: first of the height of the right, and then of the height of the left premiss of cut.  $\varphi^k, \Gamma^k$  denote  $k > 0$  occurrences of  $\varphi, \Gamma$ , respectively.

**Lemma 2 (Right reduction).** *Let  $\mathcal{D}_1 \vdash \Gamma \Rightarrow \Delta, \varphi$  and  $\mathcal{D}_2 \vdash \varphi^k, \Pi \Rightarrow \Sigma$  with  $d\mathcal{D}_1, d\mathcal{D}_2 < d\varphi$ , and  $\varphi$  principal in  $\Gamma \Rightarrow \Delta, \varphi$ , then we can construct a proof  $\mathcal{D}$  such that  $\mathcal{D} \vdash \Gamma^k, \Pi \Rightarrow \Delta^k, \Sigma$  and  $d\mathcal{D} < d\varphi$ .*

*Proof.* By induction on the height of  $\mathcal{D}_2$ . The basis is trivial, since  $\Gamma \Rightarrow \Delta, \varphi$  is identical with  $\Gamma^k, \Pi \Rightarrow \Delta^k, \Sigma$ . The induction step requires examination of all cases of possible derivations of  $\varphi^k, \Pi \Rightarrow \Sigma$ , and the role of the cut-formula in the transition. In cases where all occurrences of  $\varphi$  are parametric we simply apply the induction hypothesis to the premisses of  $\varphi^k, \Pi \Rightarrow \Sigma$  and then apply the respective rule – it is essentially due to the context independence of almost all rules and the regularity of proofs, which together prevent violation of side conditions on eigenvariables. If one of the occurrences of  $\varphi$  in the premiss(es) is a side formula of the last rule we must additionally apply weakening to restore the missing formula before the application of the relevant rule.

In cases where one occurrence of  $\varphi$  in  $\varphi^k, \Pi \Rightarrow \Sigma$  is principal we make use of the fact that  $\varphi$  in the left premiss is also principal; for the cases of contraction and weakening this is trivial. We consider the cases of lambda atoms with DD. Hence  $\mathcal{D}_1$  finishes with:

$$\frac{\Gamma \Rightarrow \Delta, \varphi[y/b] \quad \Gamma \Rightarrow \Delta, \psi[x/b] \quad \varphi[y/a], \Gamma \Rightarrow \Delta, a = b}{\Gamma \Rightarrow \Delta, (\lambda x\psi)\iota y\varphi}$$

and  $\mathcal{D}_2$  finishes with:

$$\frac{\varphi[y/a'], \psi[x/a'], (\lambda x\psi)\iota y\varphi^{k-1}, \Pi \Rightarrow \Sigma}{(\lambda x\psi)\iota y\varphi^k, \Pi \Rightarrow \Sigma}$$

or

$$\frac{(\lambda x\psi)\iota y\varphi^{k-1}, \Pi \Rightarrow \Sigma, \varphi[y/b_1] \quad (\lambda x\psi)\iota y\varphi^{k-1}, \Pi \Rightarrow \Sigma, \varphi[y/b_2] \quad b_1 = b_2, (\lambda x\psi)\iota y\varphi^{k-1}, \Pi \Rightarrow \Sigma}{(\lambda x\psi)\iota y\varphi^k, \Pi \Rightarrow \Sigma}$$

In the first case, by the induction hypothesis and lemma 1.2 we obtain  $\varphi[y/b], \psi[x/b], \Gamma^{k-1}, \Pi \Rightarrow \Delta^{k-1}, \Sigma$  and by two cuts with the leftmost and central premiss of ( $\Rightarrow \iota$ ) in  $\mathcal{D}_1$  we obtain  $\Gamma^{k+1}, \Pi \Rightarrow \Delta^{k+1}, \Sigma$ , which by contraction yields the result.

In the second case note first that by lemma 1.2 from the rightmost premiss of  $(\Rightarrow \iota)$  in  $\mathcal{D}_1$  we obtain

- a.  $\varphi[y/b_1], \Gamma \Rightarrow \Delta, b_1 = b$  and
- b.  $\varphi[y/b_2], \Gamma \Rightarrow \Delta, b_2 = b$ .

Again by the induction hypothesis from the three premisses we get:

1.  $\Gamma^{k-1}, \Pi \Rightarrow \Delta^{k-1}, \Sigma, \varphi[y/b_1]$
2.  $\Gamma^{k-1}, \Pi \Rightarrow \Delta^{k-1}, \Sigma, \varphi[y/b_2]$
3.  $b_1 = b_2, \Gamma^{k-1}, \Pi \Rightarrow \Delta^{k-1}, \Sigma$

We proceed as follows with a series of the applications of cut, followed by contractions, using the provable sequent  $b_1 = b, b_2 = b \Rightarrow b_1 = b_2$ :

$$\frac{\frac{\frac{2 \quad b}{\Gamma^k, \Pi \Rightarrow \Delta^k, \Sigma, b_2 = b} \quad \frac{\frac{1 \quad a}{\Gamma^k, \Pi \Rightarrow \Delta^k, \Sigma, b_1 = b} \quad \frac{b_1 = b, b_2 = b \Rightarrow b_1 = b_2 \quad 3}{b_1 = b, b_2 = b, \Gamma^{k-1}, \Pi \Rightarrow \Delta^{k-1}, \Sigma}}{b_2 = b, \Gamma^{2k-1}, \Pi^2 \Rightarrow \Delta^{2k-1}, \Sigma^2}}{\Gamma^{3k-1}, \Pi^3 \Rightarrow \Delta^{3k-1}, \Sigma^3}}{\Gamma^k, \Pi \Rightarrow \Delta^k, \Sigma}$$

□

**Lemma 3 (Left reduction).** *Let  $\mathcal{D}_1 \vdash \Gamma \Rightarrow \Delta, \varphi^k$  and  $\mathcal{D}_2 \vdash \varphi, \Pi \Rightarrow \Sigma$  with  $d\mathcal{D}_1, d\mathcal{D}_2 < d\varphi$ , then we can construct a proof  $\mathcal{D}$  such that  $\mathcal{D} \vdash \Gamma, \Pi^k \Rightarrow \Delta, \Sigma^k$  and  $d\mathcal{D} < d\varphi$ .*

*Proof.* By induction on the height of  $\mathcal{D}_1$  but with some important differences to the proof of the right reduction lemma. First note that we do not require  $\varphi$  to be principal in  $\varphi, \Pi \Rightarrow \Sigma$ , so it includes the case where  $\varphi$  is atomic. In all these cases we just apply the induction hypothesis. This guarantees that even if an atomic cut formula was introduced in the right premiss by  $(= -)$  the reduction of the height is achieved only on the left premiss, and we always obtain the expected result. Now, in cases where one occurrence of  $\varphi$  in  $\Gamma \Rightarrow \Delta, \varphi^k$  is principal, we first apply the induction hypothesis to eliminate all other  $k - 1$  occurrences of  $\varphi$  in the premisses and then we apply the respective rule. Since the only new occurrence of  $\varphi$  is principal, we can make use of the right reduction lemma again and obtain the result, possibly after some applications of structural rules. □

Now we are ready to prove the cut elimination theorem:

**Theorem 2.** *Every proof in GRL can be transformed into cut-free proof.*

*Proof.* By double induction: primary on  $d\mathcal{D}$  and subsidiary on the number of maximal cuts (in the basis and in the inductive step of the primary induction). We always take the topmost maximal cut and apply lemma 3 to it. By successive repetition of this procedure we reduce either the degree of a proof or the number of cuts in it until we obtain a cut-free proof. □

## 5 Adequacy

In this section, we'll make use of the fact that for every set there is a corresponding multiset, so if  $\Gamma, \Delta$  are sets of formulas, we may write  $\Gamma \Rightarrow \Delta$ . We recall that we treat  $\vee, \rightarrow, \exists$  as defined notions. For the completeness proof we assume that a denumerable set of individual constants may be added to the language.  $I$  assigns objects in the domain  $D$  of the model  $\langle D, I \rangle$  to these constants. For brevity we introduce the notation  $I_v$ , where if  $t$  is a variable or parameter,  $I_v(t) = v(t)$  and where  $t$  is a constant,  $I_v(t) = I(t)$ .

Recall the distinction between terms and pseudo-terms, the former variables and parameters and now also constants, the latter iota terms. In the following lemma,  $t$  denotes a variable, parameter or constant, not a DD, hence the proof is standard, with the case of lambda atoms similar to the case of quantifiers. In the rest of this section, too,  $t$  will refer to terms only. In particular, there is no need to consider pseudo-terms in the Lindenbaum-Henkin construction (theorem 4), because in substitution in the formulas concerned only terms can be used. Pseudo-terms are treated, just as they are in the semantics, as occurring in lambda atoms, and thus like the logical constants by the consideration of the consistent addition of formulas to a set in the construction of its maximally consistent extension.

**Lemma 4 (The Substitution Lemma.)**  $M, v \models \varphi_t^x$  iff  $M, v_{I_v(t)}^x \models \varphi$ , if  $t$  is free for  $x$  in  $\varphi$ .

*Proof.* See e.g. [7, 133f] and adjust. □

Next, the soundness of GRL.

**Theorem 3 (Soundness of GRL).** *If  $\vdash \Gamma \Rightarrow \Delta$ , then  $\models \Gamma \Rightarrow \Delta$*

*Proof.* By induction on the height of the proof. Since it is well-known that the rules of G1 are validity preserving, and it is obvious for both lambda rules, we show this property only for  $(\iota_2 \Rightarrow)$  and  $(\Rightarrow \iota)$ , leaving  $(\iota_1 \Rightarrow)$  as an exercise.

$(\iota_2 \Rightarrow)$ . Suppose (1)  $\models \Gamma \Rightarrow \Delta, \varphi_{b_1}^y$ , (2)  $\models \Gamma \Rightarrow \Delta, \varphi_{b_2}^y$ , (3)  $\models b_1 = b_2, \Gamma \Rightarrow \Delta$ , and  $\not\models (\lambda x \psi) \iota y \varphi, \Gamma \Rightarrow \Delta$ . By the last, there are a structure  $M = \langle D, I \rangle$  and assignment  $v$ , such that  $M, v \models (\lambda x \psi) \iota y \varphi$ , for all  $\gamma \in \Gamma$ ,  $M, v \models \gamma$  and for all  $\delta \in \Delta$ ,  $M, v \not\models \delta$ . Thus by (1), (2) and (3): (4)  $M, v \models \varphi_{b_1}^y$ , (5)  $M, v \models \varphi_{b_2}^y$  and (6)  $M, v \not\models b_1 = b_2$ . And there is an  $o \in D$  such that  $M, v_o^x \models \psi$ , and  $M, v_o^x \models \varphi[y/x]$ , and (7) for any  $y$ -variant  $v'$  of  $v_o^x$ , if  $M, v' \models \varphi$ , then  $v'(y) = o$ . By the conventions on the use of free and bound variables in sequents,  $x$  is not free in  $\varphi_{b_1}^y$  or  $\varphi_{b_2}^y$ , so  $v$  and  $v_o^x$  agree on them, and so by (4) and (5)  $M, v_o^x \models \varphi_{b_1}^y$  and  $M, v_o^x \models \varphi_{b_2}^y$ . By the substitution lemma,  $M, v_{o I_v(b_1)}^{x y} \models \varphi$  and  $M, v_{o I_v(b_2)}^{x y} \models \varphi$ . So the  $y$ -variants  $v'$  and  $v''$  of  $v_o^x$  that assign  $I_{v_o^x}(b_1)$  and  $I_{v_o^x}(b_2)$  to  $y$  satisfy  $\varphi$  with  $M$ , so by (7)  $I_{v'}(b_1) = I_{v''}(b_2) = o$ . But  $v'$  and  $v''$  differ from  $v$  only in what they assign to  $x$  and  $y$ , and by (6)  $I_v(b_1) \neq I_v(b_2)$ . Contradiction.

$(\Rightarrow \iota)$ . Suppose (1)  $\models \Gamma \Rightarrow \Delta, \varphi_b^y$ , (2)  $\models \Gamma \Rightarrow \Delta, \psi_b^x$ , (3)  $\models \varphi_a^y, \Gamma \Rightarrow \Delta, a = b$ , but  $\not\models \Gamma \Rightarrow \Delta, (\lambda x \psi) \iota y \varphi$ ,  $a$  not free in any formulas in  $\Gamma$  and  $\Delta$  nor in  $\varphi$ . Then

there are a structure  $M = \langle D, I \rangle$  and assignment  $v$  such that for all  $\gamma \in \Gamma$ ,  $M, v \models \gamma$ , for all  $\delta \in \Delta$ ,  $M, v \not\models \delta$  and (4)  $M, v \not\models (\lambda x\psi)iy\varphi$ . So by (1),  $M, v \models \varphi_b^y$ , by (2),  $M, v \models \psi_b^x$ , and by (4), it is not the case that there is an  $o \in D$  such that  $M, v_o^x \models \psi$ , and  $M, v_o^x \models \varphi_x^y$ , and for any  $y$ -variant  $v'$  of  $v_o^x$ , if  $M, v' \models \varphi$ , then  $v'(y) = o$ , i.e. for every  $o \in D$ , either  $M, v_o^x \not\models \psi$ , or  $M, v_o^x \not\models \varphi_x^y$ , or for some  $y$ -variant  $v'$  of  $v_o^x$ ,  $M, v' \models \varphi$  and  $v'(y) \neq o$ . Consider  $I_v(b)$ . We have either (5)  $M, v_{I_v(b)}^x \not\models \psi$ , or (6)  $M, v_{I_v(b)}^x \not\models \varphi_x^y$ , or (7) for some  $y$ -variant  $v'$  of  $v_{I_v(b)}^x$ ,  $M, v' \models \varphi$  and  $v'(y) \neq I_v(b)$ . By the substitution lemma from (5) and (6) we have  $M, v \not\models \psi_b^x$  and  $M, v \not\models \varphi_{yb}^x$ , and as  $\varphi_{xb}^y$  is the same as  $\varphi_b^y$ , this contradicts consequences of (1) and (2). By conventions on the use of free and bound variables in sequents,  $x$  and  $y$  are not free in any of their formulas, so  $v_{I_v(b)}^x$  agrees with  $v$  on all formulas in  $\Gamma$ ,  $\Delta$ , so for all  $\gamma \in \Gamma$ ,  $M, v_{I_v(b)}^x \models \gamma$ , and for all  $\delta \in \Delta$ ,  $M, v_{I_v(b)}^x \not\models \delta$ . So by (3), if  $M, v_{I_v(b)}^x \models \varphi_a^y$ , then  $M, v_{I_v(b)}^x \models a = b$ . By the substitution lemma and the semantic clause for identity, if  $M, v_{I_v(b)I_v(a)}^y \models \varphi$ , then  $I_v(a) = I_v(b)$ . Now evidently  $v_{I_v(b)I_v(a)}^y(y) = I_v(a)$ , so  $v_{I_v(b)I_v(a)}^y(y) = I_v(b)$ . But  $v_{I_v(b)I_v(a)}^y$  is a  $y$ -variant of  $v_{I_v(b)}^x$ , and the reasoning holds for any such  $y$ -variant, contradicting (7).  $\square$

Let  $\perp$  represent an arbitrary contradiction. A set of formulas  $\Gamma$  is *inconsistent* iff  $\Gamma \vdash \perp$ .  $\Gamma$  is *consistent* iff it is not inconsistent. A set of formulas  $\Gamma$  is *maximal* iff for any formula  $A$ , either  $A \in \Gamma$  or  $\neg A \in \Gamma$ . A set of formulas  $\Gamma$  is *deductively closed* iff, if  $\Gamma \vdash A$ , then  $A \in \Gamma$ . We state without proof this standard result:

**Lemma 5.** *Any maximally consistent set is deductively closed.*

Extend  $\mathcal{L}$  to a language  $\mathcal{L}^+$  by adding countably new constants ordered by a list  $\mathcal{C} = c_1, c_2, \dots$ . We will say that such a constant occurs *parametrically* if its occurrence satisfies the restrictions imposed on parameters in  $(\Rightarrow \forall)$  and  $(\iota_1 \Rightarrow)$ .

**Theorem 4.** *Any consistent set of formulas  $\Delta$  can be extended to a maximally consistent set  $\Delta^+$  such that:*

- (a) *for any formula  $\varphi$  and variable  $x$ , if  $\neg\forall x\varphi \in \Delta^+$ , then for some constant  $c$ ,  $\varphi_c^x \notin \Delta^+$ ;*
- (b) *for any formulas  $\varphi, \psi$  and variables  $x, y$ , if  $(\lambda x\psi)iy\varphi \in \Delta^+$ , then for some constant  $c$ ,  $\varphi_c^y, \psi_c^x \in \Delta^+$  and for all terms  $t$ , if  $\varphi_t^y \in \Delta^+$ , then  $t = c \in \Delta^+$ ;*
- (c) *for any formulas  $\varphi, \psi$  and variables  $x, y$ , if  $\neg(\lambda x\psi)iy\varphi \in \Delta^+$ , then for all terms  $t$ , either  $\varphi_t^y \notin \Delta^+$ , or for some constant  $c$ ,  $\varphi_c^y \in \Delta^+$  and  $c = t \notin \Delta^+$ , or  $\psi_t^x \notin \Delta^+$ .*

*Proof.* Extend  $\Delta$  by following an enumeration  $\phi_1, \phi_2, \dots$  of the formulas of  $\mathcal{L}^+$  on which every formula occurs infinitely many times as follows:

- $\Delta_0 = \Delta$
- If  $\Delta_n, \phi_n$  is inconsistent, then  $\Delta_{n+1} = \Delta_n$ .
- If  $\Delta_n, \phi_n$  is consistent, then:

(i) If  $\phi_n$  has neither the form  $\neg\forall x\varphi$  nor  $(\lambda x\psi)\iota y\varphi$  nor  $\neg(\lambda x\psi)\iota y\varphi$ , then

$$\Delta_{n+1} = \Delta_n, \phi_n.$$

(ii) If  $\phi_n$  has the form  $\neg\forall x\varphi$ , then

$$\Delta_{n+1} = \Delta_n, \neg\forall x\varphi, \neg\varphi_c^x$$

where  $c$  is the first constant of  $\mathcal{C}$  that does not occur in  $\Delta_n$  or  $\phi_n$ .

(iii) If  $\phi_n$  has the form  $(\lambda x\psi)\iota y\varphi$ , then

$$\Delta_{n+1} = \Delta_n, (\lambda x\psi)\iota y\varphi, \varphi_c^y, \psi_c^x$$

where  $c$  is the first constant of  $\mathcal{C}$  that does not occur in  $\Delta_n$  or  $\phi_n$ .

(iv) If  $\phi_n$  has the form  $\neg(\lambda x\psi)\iota y\varphi$ , then

$$\Delta_{n+1} = \Delta_n, \neg(\lambda x\psi)\iota y\varphi, \Sigma_n$$

where  $\Sigma_n$  is constructed in the following way. Take a sequence of formulas  $\sigma_1, \sigma_2, \dots$  of the form  $\varphi_{t_i}^y \rightarrow (\psi_{t_i}^x \rightarrow \neg(\varphi_c^y \rightarrow c = t))$ , where  $t$  is a term in  $\Delta_n, \phi_n$ , and  $c$  is a constant of  $\mathcal{C}$  not in  $\Delta_n, \phi_n$  or any previous formulas in the sequence. Let  $\mathcal{T} = t_1, t_2, \dots$  be an enumeration of all terms occurring in  $\Delta_n, \phi_n$ . In case  $\Delta_0$  contains infinitely many formulas, it must be ensured that  $\mathcal{C}$  is not depleted of constants needed later. So pick constants from  $\mathcal{C}$  by a method that ensures some constants are always left over for later use. The following will do. Let  $\sigma_1$  be  $\varphi_{t_1}^y \rightarrow (\psi_{t_1}^x \rightarrow \neg(\varphi_{c_1}^y \rightarrow c_1 = t_1))$ , where  $t_1$  is the first term of  $\mathcal{T}$  and  $c_1$  is the first constant of  $\mathcal{C}$  not in  $\Delta_n, \phi_n$ ; let  $\sigma_2$  be  $\varphi_{t_2}^y \rightarrow (\psi_{t_2}^x \rightarrow \neg(\varphi_{c_2}^y \rightarrow c_2 = t_2))$ , where  $t_2$  is the second term on  $\mathcal{T}$  and  $c_2$  is the  $2^2 = 4$ th constant of  $\mathcal{C}$  not in  $\Delta_n, \phi_n, \sigma_1$ . In general, let  $\sigma_n$  be  $\varphi_{t_n}^y \rightarrow (\psi_{t_n}^x \rightarrow \neg(\varphi_{c_n}^y \rightarrow c_n = t_n))$ , where  $t_n$  is the  $n$ th term of  $\mathcal{T}$  and  $c_n$  is the  $2^n$ th constant of  $\mathcal{C}$  not in  $\Delta_n, \phi_n$  nor any  $\sigma_i$ ,  $i < n$ . The entire collection of  $\sigma_i$ s is  $\Sigma_n$ .

$\Delta_{n+1}$  is consistent if  $\Delta_n, \phi_n$  is:

Case (i). Trivial.

Case (ii). Suppose  $\Delta_{n+1} = \Delta_n, \neg\forall x\varphi, \neg\varphi_c^x$  is inconsistent. Then for some finite  $\Delta'_n \subseteq \Delta_n$ :  $\vdash \Delta'_n, \neg\forall x\varphi, \neg\varphi_c^x \Rightarrow \perp$ . Hence  $\vdash \Delta'_n, \neg\forall x\varphi \Rightarrow \varphi_c^x$  by deductive properties of negation.  $c$  does not occur in any formula in  $\Delta'_n$  nor in  $\neg\forall x\varphi$ , so it occurs parametrically, and so by  $(\Rightarrow \forall)$ ,  $\vdash \Delta'_n, \neg\forall x\varphi \Rightarrow \forall x\varphi$ . Hence  $\vdash \Delta'_n \Rightarrow \forall x\varphi$ , again by deductive properties of negation. But then  $\Delta'_n, \neg\forall x\varphi$  is inconsistent, and hence so is  $\Delta_n, \neg\forall x\varphi$ .

Case (iii). Suppose  $\Delta_{n+1} = \Delta_n, (\lambda x\psi)\iota y\varphi, \varphi_c^y, \psi_c^x$  is inconsistent. Then for some finite  $\Delta'_n \subseteq \Delta_n$ ,  $\vdash \Delta'_n, (\lambda x\psi)\iota y\varphi, \varphi_c^y, \psi_c^x \Rightarrow \perp$ .  $c$  does not occur in  $\Delta'_n, (\lambda x\psi)\iota y\varphi$ , so it occurs parametrically, and hence by  $(\iota_1 \Rightarrow)$ ,  $\vdash \Delta'_n, (\lambda x\psi)\iota y\varphi \Rightarrow \perp$ , that is to say  $\Delta'_n, (\lambda x\psi)\iota y\varphi$  is inconsistent, and so is  $\Delta_n, (\lambda x\psi)\iota y\varphi$ .

Case (iv). Suppose  $\Delta_{n+1} = \Delta_n, \neg(\lambda x\psi)\iota y\varphi, \Sigma_n$  is inconsistent. Then for some finite  $\Delta'_n \subseteq \Delta_n$  and a finite  $\{\sigma_j \dots \sigma_k\} \subseteq \Sigma_n$ ,  $\vdash \Delta'_n, \neg(\lambda x\psi)\iota y\varphi, \sigma_j \dots \sigma_k \Rightarrow \perp$ . Let  $\sigma_k$  be  $\varphi_{t_k}^y \rightarrow (\psi_{t_k}^x \rightarrow \neg(\varphi_{c_k}^y \rightarrow c_k = t_k))$ . Then by the deductive properties of implication and negation:

$$\begin{aligned} &\vdash \Delta'_n, \neg(\lambda x\psi)\iota y\varphi, \sigma_j \dots \sigma_{k-1} \Rightarrow \varphi_{t_k}^y \\ &\vdash \Delta'_n, \neg(\lambda x\psi)\iota y\varphi, \sigma_j \dots \sigma_{k-1} \Rightarrow \psi_{t_k}^x \\ &\vdash \Delta'_n, \neg(\lambda x\psi)\iota y\varphi, \sigma_j \dots \sigma_{k-1}, \varphi_{c_k}^y \Rightarrow c_k = t_k \end{aligned}$$

$c_k$  was chosen so as not to occur in any previous  $\sigma_i$ ,  $i < k$ , nor in  $\Delta_n, \phi_n$ . Hence it occurs parametrically and the conditions for  $(\Rightarrow \iota)$  are fulfilled. Thus  $\vdash \Delta'_n, \neg(\lambda x\psi)\iota y\varphi, \sigma_j \dots \sigma_{k-1} \Rightarrow (\lambda x\psi)\iota y\varphi$ . But  $\vdash \Delta'_n, \neg(\lambda x\psi)\iota y\varphi, \sigma_j \dots \sigma_{k-1} \Rightarrow \neg(\lambda x\psi)\iota y\varphi$ . So  $\Delta'_n, \neg(\lambda x\psi)\iota y\varphi, \sigma_j \dots \sigma_{k-1}$  is inconsistent. Repeat this process from  $\sigma_{k-1}$  all the way down to  $\sigma_j$ , showing that  $\Delta'_n, \neg(\lambda x\psi)\iota y\varphi$  is inconsistent. Hence so is  $\Delta_n, \neg(\lambda x\psi)\iota y\varphi$ .

Let  $\Delta^+$  be the union of all  $\Delta_i$ .  $\Delta^+$  is maximal, for if neither  $\varphi$  nor  $\neg\varphi$  are in  $\Delta^+$ , then there is a  $\Delta_k \subseteq \Delta^+$  such that  $\Delta_k, \varphi \vdash \perp$  and  $\Delta_k, \neg\varphi \vdash \perp$ , but then  $\Delta_k$  is inconsistent, contradicting the method of construction of  $\Delta_k$ .  $\Delta^+$  is consistent, because otherwise some  $\Delta_i$  would have to be inconsistent, but they are not.

$\Delta^+$  satisfies (a) by construction.

To see that it satisfies (b), suppose  $(\lambda x\psi)\iota y\varphi \in \Delta^+$ . Then there is a  $\Delta_{n+1} = \Delta_n, (\lambda x\psi)\iota y\varphi, \varphi_c^y, \psi_c^x$ , and so  $\varphi_c^y, \psi_c^x \in \Delta^+$ . Suppose  $\varphi_t^y \in \Delta^+$ . Then there is a  $\Delta' \subseteq \Delta^+$  such that  $\vdash \Delta' \Rightarrow \varphi_c^y, \vdash \Delta' \Rightarrow \varphi_t^y$  and by properties of identity  $\vdash t = c \Rightarrow t = c$ . But then by  $(\iota_2 \Rightarrow)$ ,  $\vdash \Delta', (\lambda x\psi)\iota y\varphi \Rightarrow t = c$ , hence  $t = c \in \Delta^+$  by the deductive closure of  $\Delta^+$ .

To see that it satisfies (c), suppose  $\neg(\lambda x\psi)\iota y\varphi \in \Delta^+$ , but for some term  $t$ ,  $\varphi_t^y \in \Delta^+$ , (1) for all constants  $c$ , if  $\varphi_c^y \in \Delta^+$ , then  $c = t \in \Delta^+$ , and  $\psi_t^x \in \Delta^+$ . As every formula occurs infinitely many times on the enumeration of formulas of  $\mathcal{L}^+$ , there is a  $\Delta_n$  that contains  $\varphi_t^y$  and  $\psi_t^x$  and  $\Delta_{n+1} = \Delta_n, \neg(\lambda x\psi)\iota y\varphi, \Sigma_n$ . Thus  $\varphi_t^y \rightarrow (\psi_t^x \rightarrow \neg(\varphi_b^y \rightarrow b = t)) \in \Sigma_n$ , for some constant  $b$  of  $\mathcal{C}$ . Consequently, this formula is in  $\Delta^+$ , too. By the deductive properties of implication and negation and the deductive closure and consistency of  $\Delta^+$ , (2)  $\varphi_b^y \in \Delta^+$  and  $b = t \notin \Delta^+$ . But by (1) and (2),  $b = t \in \Delta^+$ . Contradiction.

This completes the proof of Theorem 4. □

**Theorem 5.** *If  $\Delta$  is a consistent set of formulas, then  $\Delta$  is satisfiable.*

*Proof.* Extend  $\Delta$  to a maximally consistent set  $\Delta^+$  as per Theorem 4. We construct a structure  $M = \langle D, I \rangle$  and function  $v: VAR \cup PAR \rightarrow D$  from  $\Delta^+$  which will satisfy  $\Delta$ .  $D$  is the set of equivalence classes of terms under identities  $t_1 = t_2 \in \Delta^+$ . Denote the equivalence class to which  $t$  belongs by  $[t]$ . For all predicate letters  $P$ ,  $\langle [t_1], \dots, [t_n] \rangle \in I(P^n)$  iff  $P^n(t_1, \dots, t_n) \in \Delta^+$ . For all variables  $v(x) = [x]$ , and for all parameters  $v(a) = [a]$ . In these latter cases  $I_v = v$ , and for all new constants of  $\mathcal{C}$ ,  $I_v(c) = [c]$ . We'll show by induction over the number of logical constants (connectives, quantifiers,  $\iota$  and  $\lambda$  symbols) in formula  $\varphi$  that  $M, v \models \varphi$  if and only if  $\varphi \in \Delta^+$ .

Suppose  $\varphi$  is an atomic formula. (a)  $\varphi$  is  $P^n(t_1, \dots, t_n)$ . Then  $M, v \models P^n(t_1, \dots, t_n)$  iff  $\langle I_v(t_1), \dots, I_v(t_n) \rangle \in I(P^n)$ , iff  $\langle [t_1], \dots, [t_n] \rangle \in I(P^n)$ , iff  $P^n(t_1, \dots, t_n) \in \Delta^+$ . (b)  $\varphi$  is  $t_1 = t_2$ . Then  $M, v \models t_1 = t_2$  iff  $I_v(t_1) = I_v(t_2)$ , iff  $[t_1] = [t_2]$ , and as these are equivalence classes under identities in  $\Delta^+$ , iff  $t_1 = t_2 \in \Delta^+$ .

For the rest of the proof suppose  $M, v \models \varphi$  if and only if  $\varphi \in \Delta$ , where  $\varphi$  has fewer than  $n$  connectives. We skip the standard cases of  $\neg, \wedge, \forall$  (see e.g. [7]).

Case 4.  $\varphi$  is  $(\lambda x\psi)t$ .

$(\lambda x\psi)t \in \Delta^+$  iff  $\psi_t^x \in \Delta^+$  by deductive closure of  $\Delta^+$ , iff  $M, v \models \psi_t^x$  by induction hypothesis.  $t$  must be free for  $x$  in  $\psi$ , hence by the substitution lemma,  $M, v \models \psi_t^x$  iff  $M, v_{I_v(t)}^x \models \psi$ , iff  $M, v_{[t]}^x \models \psi$  and  $I_v(t) = [t]$ , as the latter holds by construction of  $M$ , and this in turn is the case iff  $M, v \models (\lambda x\psi)t$  by the first semantic clause for lambda atoms.

Case 5.  $\varphi$  is  $(\lambda x\psi)iy\chi$ .

(a) If  $(\lambda x\psi)iy\chi \notin \Delta^+$ , then by deductive closure  $\neg(\lambda x\psi)iy\chi \in \Delta^+$ , and so for all terms  $t$ , either  $\chi_t^y \notin \Delta^+$ , or for some constant  $c$ ,  $\chi_c^y \in \Delta^+$  and  $c = t \notin \Delta^+$ , or  $\psi_t^x \notin \Delta^+$ .  $[t] \in D$  iff  $t$  is a term, so by induction hypothesis, for all  $[t] \in D$ , either  $M, v \not\models \chi_t^y$ , or there is a  $[c] \in D$  such that  $M, v \models \chi_c^y$  and  $M, v \not\models c = t$ , or  $M, v \not\models \psi_t^x$ .  $\chi_t^y$  is the same formula as  $\chi_{xt}^{yx}$ , so  $M, v \not\models \chi_{xt}^{yx}$ . Furthermore,  $x$  and  $y$  are not free in  $\chi_c^y$ , so for any  $o \in D$ ,  $M, v \models \chi_c^y$  iff  $M, v_o^x \models \chi_c^y$ . By the substitution lemma, either  $M, v_{I_v(t)}^x \not\models \chi_x^y$ , or  $M, v_{I_v(t)}^x \not\models \psi$ , or there is a  $[c] \in D$  such that  $M, v_{I_v(t)I_v(c)}^x \models \chi$  and  $M, v_{I_v(t)I_v(c)}^x \not\models y = x$ .  $I_v(t) = [t]$  and  $I_v(c) = [c]$ , so either  $M, v_{[t]}^x \not\models \chi_x^y$ , or  $M, v_{[t]}^x \not\models \psi$ , or there is a  $[c] \in D$  such that  $M, v_{[t][c]}^x \models \chi$  and  $M, v_{[t][c]}^x \not\models y = x$ , i.e.  $v_{[t][c]}^x(y) \neq [t]$ .  $v_{[t][c]}^x$  is a  $y$ -variant of  $v_{[t]}^x$ , hence  $M, v \not\models (\lambda x\psi)iy\chi$ .

(b) If  $(\lambda x\psi)iy\chi \in \Delta^+$ , then for some constant  $c$ ,  $\psi_c^x, \chi_c^y \in \Delta^+$  and for all terms  $t$ , if  $\chi_t^y \in \Delta^+$ , then  $c = t \in \Delta^+$ . By induction hypothesis,  $M, v \models \psi_c^x$  and  $M, v \models \chi_c^y$ . As  $y$  is either identical to  $x$  or  $x$  is not free in  $\chi$ ,  $\chi_c^y$  is the same formula as  $\chi_{xc}^{yx}$  and  $I_v(c) = [c]$ , so by the substitution lemma  $M, v_{[c]}^x \models \psi$  and  $M, v_{[c]}^x \models \chi_x^y$ . Furthermore, for all  $[t] \in D$ , if  $M, v \models \chi_t^y$ , then  $M, v \models c = t$ , i.e.  $I_v(t) = I_v(c)$ , i.e.  $I_v(t) = [c]$ . Let  $v'$  be a  $y$ -variant of  $v_{[c]}^x$ , i.e.  $v' = v_{[c][s]}^x$ , for some  $[s] \in D$ . Either  $y$  is identical to  $x$  or  $x$  is not free in  $\chi$ , so  $v_{[c][s]}^x$  and  $v$  agree on the assignments of elements of  $D$  to all variables in  $\chi$  except possibly  $y$ , and so  $M, v_{[c][s]}^x \models \chi$  iff  $M, v_{[s]}^y \models \chi$ . So suppose now  $M, v' \models \chi$  and  $v'(y) \neq [c]$ .  $v'(y) = [s]$ , so  $[c] \neq [s]$ . Then  $M, v_{[s]}^y \models \chi$ , and also if  $M, v \models \chi_s^y$ , then  $M, v \models c = s$ , i.e.  $I_v(s) = I_v(c)$ , i.e.  $I_v(s) = [c]$ . But  $I_v(s) = [s]$ , so  $I_v(s) \neq [c]$ . Hence  $M, v \not\models \chi_s^y$ , and so by the substitution lemma,  $M, v_{[s]}^y \not\models \chi$ . Contradiction.

Finally, restrict the language again to the language of  $\Delta$ : structure  $M$  constructed from  $\Delta^+$  satisfies  $\Delta$ . This completes the proof of Theorem 5.  $\square$

**Theorem 6 (Completeness for Sequents).** *If  $\models \Gamma \Rightarrow \Delta$ , then  $\vdash \Gamma \Rightarrow \Delta$ .*

*Proof.* Let  $\neg\Delta$  be the negation of all formulas in  $\Delta$ . If  $\models \Gamma \Rightarrow \Delta$ , then  $\Gamma, \neg\Delta$  is not satisfiable. Hence by Theorem 5 it is inconsistent, and as they are both finite,  $\vdash \Gamma, \neg\Delta \Rightarrow \perp$ . Hence by the properties of negation  $\vdash \Gamma \Rightarrow \Delta$ .  $\square$

**Theorem 7 (Completeness for Sets).** *If  $\Gamma \models A$ , then  $\Gamma \vdash A$ .*

*Proof.* Suppose  $\Gamma \models A$ . Then  $\Gamma, \neg A$  is not satisfiable, hence by Theorem 5 it is inconsistent and  $\Gamma, \neg A \vdash \perp$ . So for some finite  $\Sigma \subseteq \Gamma, \neg A$ ,  $\Sigma \Rightarrow \perp$ . If  $\neg A \in \Sigma$ , then by the deductive properties of negation,  $\Sigma - \{\neg A\} \Rightarrow A$ , and as  $\Sigma - \{\neg A\}$  is certain to be a subset of  $\Gamma$ ,  $\Gamma \vdash A$ . If  $\neg A \notin \Sigma$ , then  $\Sigma \Rightarrow A$  by the properties of negation, and again  $\Gamma \vdash A$ .  $\square$



By theorem 1 and 7 we also obtain the (strong) completeness of HRL.

## 6 Conclusion

Summing up, **RL** saves the essential features of the Russellian approach to definite descriptions. It avoids problems like the arbitrary restriction of axiom  $R$  to predicate symbols and scoping difficulties. In the semantics it retains the reductionist Russellian flavour in the sense that DD are not characterised by an interpretation function, but instead they are treated as a case in the clauses of the forcing definition for lambda atoms. In this respect **RL** is different from the approach provided by Fitting and Mendelsohn [10] which is closer to the Fregean tradition.

The rules of GRL are in principle direct counterparts of the tableau rules from [17] but with two important exceptions. The tableau rule corresponding to  $(= -)$  is not restricted to atomic formulas and the tableau rule corresponding to  $(\iota_2 \Rightarrow)$  is not branching. Its counterpart in sequent calculus would be:

$$(\iota_2 \Rightarrow') \frac{b_1 = b_2, \Gamma \Rightarrow \Delta}{(\lambda x \psi) \iota_2 y \varphi, \varphi[y/b_1], \varphi[y/b_2], \Gamma \Rightarrow \Delta}$$

Such a non-branching rule is certainly much better for proof search, but it is not possible to prove the cut elimination theorem in its presence. The same applies to  $(= -)$  without restriction to atomic formulas. In both cases the occurrences of arbitrary formulas  $\varphi$  in the antecedent of the conclusion can be cut formulas and, in case the cut formula in the left premiss of the cut application is principal, it is not possible to make a reduction of the complexity of the cut formulas.

There is an interesting advantage of introducing the sequent characterisation of **RL** over tableau formalisation from [17]. Since no rule specific to GRL has more than one active formula in the succedent they are also correct in the setting of intuitionistic logic as characterised by G1i [34]. It is sufficient to change the background calculus for the intuitionistic version (with  $(\leftrightarrow \Rightarrow)$ ,  $(\Rightarrow \vee)$  split into two rules, and  $(\Rightarrow C)$ ,  $(\Rightarrow W)$  deleted) and check that all proofs from section 3, 4 hold also for a (syntactically characterised) intuitionistic version of **RL**. By comparison, the changes in the tableau setting would be rather more involved and connected with the introduction of labels for naming the states of knowledge in the constructed model.

The approach provided here may be modified also to cover some more expressive logics (like modal ones) and some other theories of DD like those proposed in the context of free logics. Some preliminary work in this direction is found in [12] and [13]. On the other hand the problems briefly mentioned in section 1 need serious examination and this may be carried out only after the implementation of the presented formal systems. This is one of the most important future tasks.

*Acknowledgements.* We would like to thank Michał Zawidzki for his comments and suggestions.

## References

1. Artale, A., Mazzullo, A., Ozaki, A., Wolter, F.: On Free Description Logics with Definite Descriptions. In: Proceedings of the 18th International Conference on Principles of Knowledge Representation and Reasoning (pp. 63–73). IJCAI Organization. (2021)
2. Beeson, M.: Foundations of Constructive Mathematics. Springer (1985)
3. Benzmüller, C., Scott, D. S.: Automating Free Logic in HOL, with an Experimental Application in Category Theory. *Journal of Automated Reasoning* **64**: 53–72 (2020)
4. Blumson, B.: Anselm’s God in Isabelle/HOL URL: [https://www.isa-afp.org/browser\\_info/current/AFP/AnselmGod/document.pdf](https://www.isa-afp.org/browser_info/current/AFP/AnselmGod/document.pdf) (2020)
5. Bohrer, B., Fernández, M., Platzer, A.:  $dL_l$ : Definite Descriptions in Differential Dynamic Logic. In: Fontaine, Pascal (ed.): Automated Deduction – CADE 27. Springer International Publishing, pp. 94–110 (2019)
6. Bressan, A., A General Interpreted Modal Calculus. Yale University. Press, Yale (1972)
7. Enderton, H. B.: A Mathematical Introduction to Logic. Harcourt Academic Press, San Diego (2000)
8. Feferman, S.: Definedness. *Erkenntnis* **43**, 295–320 (1995)
9. Fitting, M.: A Modal Logic Epsilon-Calculus. *Notre Dame Journal of Formal Logic* **16**(1), 1–16 (1975)
10. Fitting, M., Mendelsohn, R. L.: First-Order Modal Logic. Synthese Library, vol. 277, Springer, Dordrecht (1998)
11. Francez, N., Więckowski, B.: A proof-theoretic semantics for contextual definiteness. In: Moriconi, E, Tesconi, L. (eds.). Proceedings of the Second Pisa Colloquium in Logic, Language and Epistemology. Edizioni ETS, Pisa, 181–212 (2014)
12. Indrzejczak, A.: Existence, definedness and definite descriptions in hybrid modal logic. In: Olivetti, N., Verbrugge, R., Negri, S., Sandu, G. (eds.) *Advances in Modal Logic* 13, pp. 349–368, College Publications, Rickmansworth (2020)
13. Indrzejczak, A.: Free Definite Description Theory – Sequent Calculi and Cut Elimination. *Logic and Logical Philosophy* **29**(4) 505–539 (2020)
14. Indrzejczak, A.: Free Logics are Cut-Free. *Studia Logica*. **109** 859–886 (2021)
15. Indrzejczak, A.: Sequents and Trees. An Introduction to the Theory and Applications of Propositional Sequent Calculi. Birkhäuser (2021)
16. Indrzejczak, A.: Russellian definite description theory—a proof-theoretic approach. *The Review of Symbolic Logic*. **16**(2), 624–649 (2023)
17. Indrzejczak, A., Zawidzki, M.: When Iota meets Lambda. *Synthese*. **201**(2), 1–33 (2023)
18. Kalish, D., Montague, R., Mar, G.: Logic. Techniques of Formal Reasoning (2 ed.). Oxford University Press, New York, Oxford (1980)
19. Kürbis, N.: A Binary Quantifier for Definite Descriptions in Intuitionist Negative Free Logic: Natural Deduction and Normalization. *Bulletin of the Section of Logic* **48**(2): 81–97 (2019)
20. Kürbis, N.: Two Treatments of Definite Descriptions in Intuitionist Negative Free Logic. *Bulletin of the Section of Logic* **48**(4): 299–317 (2019)
21. Kürbis, N.: Definite Descriptions in Intuitionist Positive Free Logic. *Logic and Logical Philosophy* **20**(2): 327–358 (2021)
22. Kürbis, N.: Proof-Theory and Semantics for a Theory of Definite Descriptions. In: Das, A., Negri, S. (eds.) *Automated Reasoning with Analytic Tableaux and Related Methods*. Springer, Berlin, Heidelberg (2021)

23. Kürbis, N.: A Binary Quantifier for Definite Descriptions for Cut Free Free Logics. *Studia Logica* **110**(1): 219–239 (2022)
24. Metcalfe, G., Olivetti, N., Gabbay, D.: *Proof Theory for Fuzzy Logics*. Springer (2008)
25. Neuhaus, F., Kutz, O., Righetti, G.: Free description logic for ontologists. In: Hammar, K. et al. (Eds.), *Proceedings of the Joint Ontology Workshops co-located with the Bolzano Summer of Knowledge (BOSK 2020)* Vol. 2708. Bozen-Bolzano (2020)
26. Oppenheimer, P. E., Zalta, E. N.: A Computationally-Discovered Simplification of the Ontological Argument. *Australasian Journal of Philosophy* **89**: 333–349 (2011)
27. Orlandelli, E.: Labelled calculi for quantified modal logics with definite descriptions. *Journal of Logic and Computation* **31**(3) 923–946 (2021)
28. Russell, B.: On Denoting. *Mind* **XIV** 479–494 (1905)
29. Scales, R.: *Attribution and Existence*. Ph.D. Dissertation. University of California, Irvine (1969)
30. Scott, D.: Identity and Existence in Intuitionistic Logic. In: M. Fourman, C. Mulvey and D. Scott (eds.) *Applications of Sheaves*. Springer pp.660–696 (1979)
31. Stalnaker, R. C., Thomason, R. H.: Abstraction in First-Order Modal Logic. *Theoria* **34**(3) 203–207 (1968)
32. Tennant, N.: A General Theory of Abstraction Operators. *The Philosophical Quarterly* **54**(214) 105–133 (2004)
33. Thomason, R. H.: Some Completeness Results for Modal Predicate Calculi. In: Lambert, K. (ed.) *Philosophical Problems in Logic*. Reidel pp.56–76 (1970)
34. Troelstra, A. S., Schwichtenberg, H.: *Basic Proof Theory*. Oxford University Press, Oxford (1996)
35. Whitehead, A. N., Russell, B.: *Principia Mathematica* vol. I. Cambridge University Press, Cambridge (1910)