Existence, Definedness and Definite Descriptions in Hybrid Modal Logic

Andrzej Indrzejczak [1]

Department of Logic, University of Łódź
Lindleya 3/5, 90–131 Łódź
e-mail:andrzej.indrzejczak@filozof.uni.lodz.pl

Abstract

The paper presents a sequent calculus HFM for first-order hybrid modal logic with lambda operator, existence and definedness predicates. It is particularly useful for dealing with non-rigid and non-designating terms and the apparatus of hybrid logics provides a satisfactory structural proof theory of this logic. Its reduct is shown to be equivalent to Fitting and Mendelsohn's tableau system for first-order modal logic by series of syntactical transformations. Additionally, some account of definite descriptions is formulated in terms of extended calculus HFMD and the whole system satisfies cut elimination theorem.

Keywords: first-order modal logic, hybrid modal logic, definite descriptions, sequent calculus, cut elimination.

1 Introduction

First-Order Modal Logic (FOML) is a field far from commonly accepted solutions. During the last two decades at least three important books due to Fitting and Mendelsohn [10], Garson [11], and Goldblatt [13] provided a detailed treatment of different solutions to philosophical and technical problems connected with FOML. One can also find practically useful deductive systems working even for very sophisticated systems. Prefixed tableaux in [10] and natural deduction (ND) in [11] are good examples, yet, they do not provide well-behaved formulations in the sense of structural proof theory. Recently, the work of Corsi and Orlandelli [25], and Orlandelli [24], provide satisfactory proof-theoretic approach in the framework of labelled sequent calculi (SC). This raises the question if more standard version of SC may be used for that aim. In [16] we provided a standard SC for Garson's version of FOML. In this paper we want to focus on the more demanding approach of Fitting and Mendelsohn (FM) and provide for it a standard version of SC satisfying cut elimination.

[1] The results reported in this paper are supported by the National Science Centre, Poland (grant number: DEC-2017/25/B/HS1/01268).

There are at least two important features which make FM one of the most subtle and expressive logics, providing satisfactory solutions to several philosophical problems involved in FOML. The first feature is connected with the application of predicate abstracts; the second with paying attention to the distinction between existence and definedness. It leads to more flexible treatment of scoping difficulties, of non-rigid and non-denoting terms. Both features are expressible in semantics but the resulting logic is hardly representable in a standard axiomatic way. Therefore, on the level of proof systems (tableaux in this case), additional devices are introduced like double prefixing – of formulae and of terms. This may be seen as the third original feature of their approach although not of the logic but of its tableau presentation. Let us comment on these three features and their significance for our proof-theoretic study.

Predicate abstracts built by means of the lambda operator were introduced to studies on FOML by Thomason and Stalnaker [29] and then the technique was developed by Fitting [9]. In the realm of modal logic this technique is mainly used for taking control over scoping difficulties concerning modal operators but also complex terms like definite descriptions. From the standpoint of proof theory it has additional advantages. In general, introduction of complex terms leads to serious problems connected with unrestricted instantiation of such terms for variables. A freedom of instantiation in quantifier rules usually destroys the subformula property and cut elimination proof. The application of predicate abstracts opens a way to avoid such problems by separation of a direct predication restricted to variables and indirect predication via lambda operators. In consequence, respective rules for quantifiers may be restricted to variables as the only allowed instantiated terms without losing generality.

A distinction between existence and definedness (or denotation) usually goes unnoticed although, as is firmly emphasized in [10] "these are really orthogonal issues. Terms designate; objects exist." It is worth noting that the separation of these notions is also important in studies on constructive mathematics and applications to computer science; see for example Beeson [1] and Fefermann [8]. True, this difference may be easily lost if existence is defined as $\exists x(x = t)$ and definedness as $t = t$ since in the context of negative free logic (NFL) both formulae are provably equivalent. Hence, at least in NFL, this conceptual difference cannot be syntactically represented in a sensible way. However, in FM the application of a richer apparatus enables a syntactical separation of these notions. As a consequence, in the proposed language we can talk about existent and non-existent objects, as well as denoting and non-denoting terms. Suitable predicates are definable in FM; in the system proposed below we introduce them as primitive notions which facilitates syntactic control.

The application of prefixes denoting possible worlds and encoding accessibility relations between them is a well known technique due to Fitting and applied usually to formulae for stating that they hold in respective worlds. In FM prefixes are additionally linked to terms to signify their denotation in worlds denoted by prefixes. In our approach we use this solution not as an auxiliary technical device but as a part of our language. More specifically, we will use a variant of hybrid logic (HL) with twofold application of satisfaction-operators; to formulae and to terms. In contrast to FM we restrict this kind of "rigidification" only to terms which are not variables. Variables in FM are rigid by definition and addition of prefixes is not necessary from this point of view. However, in [10] this technique is applied to variables for enabling control over actualist quantifiers without the need of explicit application of the existence predicate. In our system, the existence predicate is primitive and there is no need to overload variables with unnecessary decoration.

The application of HL to provide well-behaved proof-theoretic representation of FM is not accidental. HL is an interesting generalization of standard modal logic with well established body of general results and extremely rich syntactic resources. The basic language of HL is obtained by the addition of the second sort of propositional atoms called nominals. Informally, they denote propositions true in exactly one world of a model and may serve as names of these worlds. Additionally one can introduce several specific operators; the most important are satisfaction, or "at"-operators, indicating that a formula is satisfied in the world denoted by some nominal. This permits for internalization of the essential part of semantics in the language. What is nice with HL is the fact that changes in the language do not affect seriously the rest of the machinery applied in standard modal logic. In particular, modifications in the relational semantics are minimal. The concept of a frame remains intact, only on the level of models we have some changes. These relatively small modifications of standard modal languages give us many advantages: more expressive language, better behavior in completeness theory, more natural and simpler proof theory. In particular, one may define in HL such frame conditions like irreflexivity, asymmetry, trichotomy and others not expressible in standard modal languages. Proof theory of HL, developed in the framework of tableaux or natural deduction offers even more general approach than application of labels popular in proof theory for standard modal logic.

The aim of the paper is twofold: 1 Extension of HL to obtain fuller expressibility of phenomena so far dealt with only in standard FOML. 2 Providing well-behaved structural proof theory for FM. The original FM is well defined semantically and by means of tableaux which are useful in practice but not fully satisfactory from the theoretical standpoint. Many significant features are introduced as additional technical devices or left implicit (like clauses concerning definedness of terms and existence of objects). What we gain is an SC where all this stuff is introduced explicitly and treated in a uniform fashion by means of well-behaved symmetric and analytic rules satisfying cut elimination.

We start with a brief account of the language and semantics of HL. In section 3 a system HFM (for Hybrid FM) will be presented. Its adequacy is shown indirectly in stages in section 4. First by translation of Fitting's and Mendelsohn's tableau (FM-T) proofs into proofs in some auxiliary calculus HFM1. Then by showing that every proof in HFM1 is simulated in another system HFM2 and vice versa. Finally that HFM2 is equivalent to a reduct of HFM. In section 5 we will extend HFM to cover definite descriptions.

2 Preliminaries

In what follows we assume a standard predicate monomodal language with denumerable sets of predicate symbols $PRED$ (symbolised with P) and function symbols FUN (symbolised with f), both of any arity $n \geq 0$. Incidentally, for representing 0-ary functions we will use c (for individual constant). Individual variables are divided into disjoint sets of bound and free variables (respectively VAR represented by $x, y, z, \ldots$ and PAR, for parameters, represented by $a, b, \ldots$). The set of logical constants comprises boolean and modal connectives: $\neg, \wedge, \vee, \rightarrow, \Box, \Diamond$, (actualist) quantifiers: $\exists, \forall$ and identity predicate $=$. To this basic assortment we add special constants from FM: unary predicates of existence E, nonexistence E^-, definedness D, binary term equality $\approx$, and lambda operator λ for forming predicate abstracts. Categories of terms $TERM$ and formulae FOR are defined in a standard way with an additional clause for the lambda operator:

- if $\varphi \in FOR$ and $t \in TERM$, then $(\lambda x\varphi)t \in FOR$

Hybrid version of this language is obtained by addition of two components: a denumerable set of propositional symbols called nominals $NOM = \{i, j, k, ...\}$ and a denumerable set of unary satisfaction operators (sat-operators) indexed by nominals $@_i$. Following Blackburn and Marx [3] (see also [4], [22], [21]) we will use the latter in two functions; the new clauses are:

- if $\varphi \in FOR$ and $i \in NOM$, then $@_i\varphi \in FOR$
- if $t \in TERM$ and $i \in NOM$, then $@_it \in TERM$

The first reads "φ is satisfied in a state i". The second – "$@_it$ names a designatum of t in a state i". For the language of the basic system HFM we restrict the second application of sat-operators to terms other than parameters.

Nominals are introduced for naming states (worlds) of a model domain so in a sense they are terms. However syntactically they are treated as ordinary sentences. In particular, they can be combined by means of boolean and modal connectives. Informally they represent propositions "the name of the actual state is i". On the other hand, they are just names of states when they occur as indices of sat-operators. It is important to note that both nominals and sat-operators are genuine language elements not an extra metalinguistic machinery as in several kinds of labelled systems.

The notion of a frame is defined as for standard FOML and a model is any structure $\mathfrak{M} = \langle \mathcal{W}, \mathcal{R}, \mathcal{D}, d, \mathcal{I}, \mathcal{I}_w \rangle$, where $\mathcal{W}, \mathcal{R}$ is a standard modal frame, $\mathcal{D}$ is a nonempty domain, $d : \mathcal{W} \longrightarrow \mathcal{P}(\mathcal{D})$ is a function which assigns a (nonempty) set of (existent) objects to every world, $\mathcal{I}(i) \in \mathcal{W}$ for every nominal i, and $\mathcal{I}_w$ is a family of world's relative functions of interpretation, defined as follows:

$\mathcal{I}_w(P^n) \subseteq \mathcal{D}^n$, for every n-argument predicate and world;

$\mathcal{I}_w(c) \in \mathcal{D}$, if defined;

$\mathcal{I}_w(f^n) \in \mathcal{D}^{\mathcal{D}^n}$, if defined.

Note that in the last two cases different members of $\mathcal{D}$ and different functions may be selected as designates of c, f in different worlds, so terms (other than variables) are generally non-rigid. Moreover $\mathcal{I}_w$ is partial, i.e. for some w it may be not defined. In case of individual names it means that in some worlds (possibly all) they may be non-denoting. For function symbols it means that corresponding functions are partial, i.e. defined on subsets of $\mathcal{D}^n$.

An assignment v is defined in a standard way as $v : VAR \cup PAR \longrightarrow \mathcal{D}$, hence parameters are rigid terms by definition. An x-variant v' of v is like v with possibly $v'(x) \neq v(x)$; we will use a common notation v_0^x for x-variant of v with specified value of x. Interpretation $\mathcal{I}_w^v(t)$ of a term t in w under an assignment v is just $v(t)$ for elements of VAR and PAR, $\mathcal{I}_w(t)$ for $t \in FUN$. Hence, $\mathcal{I}_w^v(c) = \mathcal{I}_w(c)$, if it is defined; $\mathcal{I}_w^v(f^n(t_1, ..., t_n)) = \mathcal{I}_w(f^n)\langle\mathcal{I}_w^v(t_1), ..., \mathcal{I}_w^v(t_n)\rangle$, if each $\mathcal{I}_w^v(t_i)$ is defined, and $\langle\mathcal{I}_w^v(t_1), ..., \mathcal{I}_w^v(t_n)\rangle$ is in the domain of $\mathcal{I}_w(f^n)$. In case $t := @_i t'$ it is $\mathcal{I}_{\mathcal{I}(i)}^v(t')$. Hence, attaching the sat-operator $@_i$ to term t' makes it a rigid term, namely an object designed by t' in $\mathcal{I}(i)$ (if there is such an object). That is why it does not make sense to apply sat-operators to parameters; they are already rigid terms. On the other hand in case of complex term $f^n t_1, ..., t_n$ it is not enough to add $@_i$ as a prefix to make it rigid; all arguments must be rigid. In what follows we will use r to denote any rigid term – a parameter or a term with sat-operators attached to all nonparametric components. The clauses for the satisfaction relation are defined as follows:

$\mathfrak{M}, v, w \models P^n(r_1, ..., r_n)$	iff	$\langle\mathcal{I}_w^v(r_1), ..., \mathcal{I}_w^v(r_n)\rangle \in \mathcal{I}_w(P^n)$ and $\mathcal{I}_w^v(r_i) \in \mathcal{D}, i \leq n$
$\mathfrak{M}, v, w \models r_1 = r_2$	iff	$\mathcal{I}_w^v(r_1) = \mathcal{I}_w^v(r_2)$ and $\mathcal{I}_w^v(r_i) \in \mathcal{D}, i \leq 2$
$\mathfrak{M}, v, w \models t_1 \approx t_2$	iff	$\mathcal{I}_w^v(t_1) = \mathcal{I}_w^v(t_2)$ and $\mathcal{I}_w^v(t_i) \in \mathcal{D}, i \leq 2$
$\mathfrak{M}, v, w \models Et$	iff	$\mathcal{I}_w^v(t) \in d(w)$
$\mathfrak{M}, v, w \models Dt$	iff	$\mathcal{I}_w^v(t) \in \mathcal{D}$
$\mathfrak{M}, v, w \models \neg\varphi$	iff	$\mathfrak{M}, v, w \not\models \varphi$
$\mathfrak{M}, v, w \models \varphi \to \psi$	iff	$\mathfrak{M}, v, w \not\models \varphi$ or $\mathfrak{M}, v, w \models \psi$
$\mathfrak{M}, v, w \models \Box\varphi$	iff	$\mathfrak{M}, v, w' \models \varphi$ for any w' such that $\mathcal{R}ww'$
$\mathfrak{M}, v, w \models \Diamond\varphi$	iff	$\mathfrak{M}, v, w' \models \varphi$ for some w' such that $\mathcal{R}ww'$
$\mathfrak{M}, v, w \models \forall x\varphi$	iff	$\mathfrak{M}, v_o^x, w \models \varphi$ for all $o \in d(w)$
$\mathfrak{M}, v, w \models \exists x\varphi$	iff	$\mathfrak{M}, v_o^x, w \models \varphi$ for some $o \in d(w)$
$\mathfrak{M}, v, w \models (\lambda x\varphi)t$	iff	$\mathcal{I}_w^v(t) \in \mathcal{D}$ and $\mathfrak{M}, v_o^x, w \models \varphi$ for $o = \mathcal{I}_w^v(t)$
$\mathfrak{M}, v, w \models i$	iff	$w = \mathcal{I}(i)$
$\mathfrak{M}, v, w \models @_i\varphi$	iff	$\mathfrak{M}, v, w' \models \varphi$, where $w' = \mathcal{I}(i)$

Note that first two clauses are restricted to rigid terms. Also the only difference between $=$ and $\approx$ is that the latter is defined for all terms. In the original FM semantics [10] atomic formulae and equalities with $=$ are restricted only to variables and have a simpler form:

$\mathfrak{M}, v, w \models P^n(x_1, ..., x_n)$	iff	$\langle v(x_1), ..., v(x_n)\rangle \in \mathcal{I}_w(P^n)$
$\mathfrak{M}, v, w \models x_1 = x_2$	iff	$v(x_1) = v(x_2)$

Rigid terms (other than variables) as arguments of predicates are admissible only as a technical device in FM tableaux. Predicates $E, E^-, D, \approx$ are treated similarly as defined notions. In fact, we could dispense with $\approx$ in FM but it will be necessary later for the treatment of definite descriptions in section 5.

Definitions of truth in a model, satisfiability, validity and entailment are standard. We obtain different normal modal logics by restricting $\mathcal{R}$ suitably.

3 Sequent Calculus HFM

Various proof systems for different hybrid logics were constructed, including tableaux (Blackburn [2], Blackburn and Marx [3], Zawidzki [30]) and natural deduction (ND) (Braüner [6], Indrzejczak [14]). Most of them represent so called sat-calculi where each formula is preceded by the sat-operator. Using sat-calculi instead of calculi working with arbitrary formulae is justified by the fact that φ holds in (any) HL iff $@_i\varphi$ holds, provided i is not present in φ. So a proof of $@_i\varphi$ is equivalent to a proof of φ. One may find several cut-free sat-SC for some HL in different languages independently proposed by Blackburn [2], Braüner [6], Bolander and Braüner [5], Indrzejczak and Zawidzki [19]. In all these cases SC are obtained by translation; either from tableaux or from (normalizable) ND. Hence these systems are cut-free but with no direct syntactical proof for cut elimination. A constructive cut elimination proof for propositional sat-SC was provided by Indrzejczak [15]. In what follows we will use an extended form of this calculus. It consists of the following rules which we divide into several groups for easier reference. Sequents are composed from finite multisets of sat-formulae of the form $@_i\varphi$.

1. Structural rules:

$$(AX)\ \ @_i\varphi \Rightarrow @_i\varphi \quad (C\Rightarrow)\ \frac{@_i\varphi, @_i\varphi, \Gamma \Rightarrow \Delta}{@_i\varphi, \Gamma \Rightarrow \Delta} \quad (\Rightarrow C)\ \frac{\Gamma \Rightarrow \Delta, @_i\varphi, @_i\varphi}{\Gamma \Rightarrow \Delta, @_i\varphi}$$

$$(W\Rightarrow)\ \frac{\Gamma \Rightarrow \Delta}{@_i\varphi, \Gamma \Rightarrow \Delta} \quad (\Rightarrow W)\ \frac{\Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, @_i\varphi} \quad (Cut)\ \frac{\Gamma \Rightarrow \Delta, @_i\varphi \qquad @_i\varphi, \Pi \Rightarrow \Sigma}{\Gamma, \Pi \Rightarrow \Delta, \Sigma}$$

2. Propositional (Boolean) rules:

$$(\neg\Rightarrow)\ \frac{\Gamma \Rightarrow \Delta, @_i\varphi}{@_i\neg\varphi, \Gamma \Rightarrow \Delta} \qquad (\Rightarrow\neg)\ \frac{@_i\varphi, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, @_i\neg\varphi}$$

$$(\wedge\Rightarrow)\ \frac{@_i\varphi, @_i\psi, \Gamma \Rightarrow \Delta}{@_i(\varphi\wedge\psi), \Gamma \Rightarrow \Delta} \qquad (\Rightarrow\wedge)\ \frac{\Gamma \Rightarrow \Delta, @_i\varphi \qquad \Gamma \Rightarrow \Delta, @_i\psi}{\Gamma \Rightarrow \Delta, @_i(\varphi\wedge\psi)}$$

$$(\Rightarrow\rightarrow)\ \frac{@_i\varphi, \Gamma \Rightarrow \Delta, @_i\psi}{\Gamma \Rightarrow \Delta, @_i(\varphi\rightarrow\psi)} \qquad (\rightarrow\Rightarrow)\ \frac{\Gamma \Rightarrow \Delta, @_i\varphi \qquad @_i\psi, \Gamma \Rightarrow \Delta}{@_i(\varphi\rightarrow\psi), \Gamma \Rightarrow \Delta}$$

3. Modal basic rules:

$$(\Rightarrow\Box)\ \frac{@_i\Diamond j, \Gamma \Rightarrow \Delta, @_j\varphi}{\Gamma \Rightarrow \Delta, @_i\Box\varphi} \qquad (\Box\Rightarrow)\ \frac{\Gamma \Rightarrow \Delta, @_i\Diamond j \qquad @_j\varphi, \Gamma \Rightarrow \Delta}{@_i\Box\varphi, \Gamma \Rightarrow \Delta}$$

$$(\Diamond\Rightarrow)\ \frac{@_i\Diamond j,\ @_j\varphi, \Gamma \Rightarrow \Delta}{@_i\Diamond\varphi, \Gamma \Rightarrow \Delta} \qquad (\Rightarrow\Diamond)\ \frac{\Gamma \Rightarrow \Delta, @_i\Diamond j \qquad \Gamma \Rightarrow \Delta, @_j\varphi}{\Gamma \Rightarrow \Delta, @_i\Diamond\varphi}$$

where $\varphi \notin NOM$, j does not occur in the conclusion of $(\Rightarrow\Box), (\Diamond\Rightarrow)$.

4. Nominal rules:

These rules are specific for HL and mainly connected with the fact that nominals are formulae of the language, not external devices like prefixes and labels in external proof systems.

$$(@\Rightarrow)\ \frac{@_i\varphi, \Gamma\Rightarrow\Delta}{@_j@_i\varphi,\ \Gamma\Rightarrow\Delta} \qquad (\Rightarrow @)\ \frac{\Gamma\Rightarrow\Delta, @_i\varphi}{\Gamma\Rightarrow\Delta,\ @_j@_i\varphi} \qquad (Ref)\ \frac{@_i i, \Gamma\Rightarrow\Delta}{\Gamma\Rightarrow\Delta}$$

$$(Nom_1)\ \frac{@_i\Diamond k, \Gamma\Rightarrow\Delta}{@_i j, @_j\Diamond k, \Gamma\Rightarrow\Delta} \qquad (Nom_2)\ \frac{@_i\varphi, \Gamma\Rightarrow\Delta}{@_i j, @_j\varphi, \Gamma\Rightarrow\Delta}$$

where φ is atomic or nominal in (Nom_2).

5. Modal frame rules

Rules 1–4 provide an adequate HL formalization of K. In order to cover stronger logics adequate with respect to suitably restricted classes of frames one must add special rules for frame conditions. It may be done in a uniform fashion for many logics by means of standard hybrid translation HT from first-order language into basic hybrid language defined as follows:

$$\begin{array}{lcllcl} HT(Rtt') & = & @_t\Diamond t' & HT(Pt) & = & @_t p \\ HT(t=t') & = & @_t t' & HT(\neg\varphi) & = & \neg HT(\varphi) \\ HT(\varphi\vee\psi) & = & HT(\varphi)\vee HT(\psi) & HT(\exists u\varphi) & = & \exists u HT(\varphi) \end{array}$$

Braüner [6] states that for every basic geometric formula of the form:

$\forall x_1, ..., x_k(\varphi_1\wedge ...\wedge\varphi_n\to\exists y_1, ..., y_l(\psi_1\vee ...\vee\psi_m))$,

where $k\geq 1, l, n, m\geq 0$, each φ_i is an atom and each ψ_i is an atom or finite conjunction of atoms there corresponds a frame rule (FR) of the form:

$$\frac{\Gamma\Rightarrow\Delta, \varphi'_1\quad ...\quad \Gamma\Rightarrow\Delta, \varphi'_n\qquad \Psi_1, \Gamma\Rightarrow\Delta\quad ...\quad \Psi_m, \Gamma\Rightarrow\Delta}{\Gamma\Rightarrow\Delta}$$

where $k\geq 1, l, n, m\geq 0$, each $\varphi'_i = HT(\varphi_i)$, each Ψ_i is a set of HT-translations of atoms that form conjunction ψ_i and no nominal that corresponds to y_i occurs in $\Gamma_1-\Gamma_m, \Delta, \varphi'_1-\varphi'_n$.

6. Specific HFM rules:

$$(\forall E\Rightarrow)\ \frac{\Gamma\Rightarrow\Delta, @_iEb\quad @_i\varphi[x/b], \Gamma\Rightarrow\Delta}{@_i\forall x\varphi, \Gamma\Rightarrow\Delta} \qquad (\Rightarrow\forall E)\ \frac{@_iEa, \Gamma\Rightarrow\Delta, @_i\varphi[x/a]}{\Gamma\Rightarrow\Delta, @_i\forall x\varphi}$$

$$(\Rightarrow\exists E)\ \frac{\Gamma\Rightarrow\Delta, @_iEb\quad \Gamma\Rightarrow\Delta, @_i\varphi[x/b]}{\Gamma\Rightarrow\Delta, @_i\exists x\varphi} \qquad (\exists E\Rightarrow)\ \frac{@_iEa, @_i\varphi[x/a], \Gamma\Rightarrow\Delta}{@_i\exists x\varphi, \Gamma\Rightarrow\Delta}$$

$$(E-)\ \frac{@_iEa, \Gamma\Rightarrow\Delta}{\Gamma\Rightarrow\Delta} \qquad (D-)\ \frac{@_iDb, \Gamma\Rightarrow\Delta}{\Gamma\Rightarrow\Delta} \qquad (\approx -)\ \frac{@_it\approx t, \Gamma\Rightarrow\Delta}{@_iDt,\ \Gamma\Rightarrow\Delta}$$

$$(=+)\ \frac{@_i\varphi[x/r_1], \Gamma\Rightarrow\Delta}{@_j r_1 = r_2, @_i\varphi[x/r_2], \Gamma\Rightarrow\Delta} \quad \text{with } \varphi \text{ atomic formula.}$$

$$(\lambda\Rightarrow)\ \frac{@_i Dt, @_i\varphi[x/t^{@_i}], \Gamma\Rightarrow\Delta}{@_i(\lambda x\varphi)t, \Gamma\Rightarrow\Delta} \qquad (\Rightarrow\lambda)\ \frac{\Gamma\Rightarrow\Delta, @_i Dt \quad \Gamma\Rightarrow\Delta, @_i\varphi[x/t^{@_i}]}{\Gamma\Rightarrow\Delta, @_i(\lambda x\varphi)t}$$

$$(\Rightarrow E)\ \frac{\Gamma\Rightarrow\Delta, @_i Dt \quad \Gamma\Rightarrow\Delta, @_i Eb \quad \Gamma\Rightarrow\Delta, @_i t^{@_i} = b}{\Gamma\Rightarrow\Delta, @_i Et}$$

$$(E\Rightarrow)\ \frac{@_i Dt, @_i Ea, @_i t^{@_i} = a, \Gamma\Rightarrow\Delta}{@_i Et, \Gamma\Rightarrow\Delta}$$

$$(E^-\Rightarrow)\ \frac{@_i Dt, \Gamma\Rightarrow\Delta, @_i Eb \quad @_i Dt, \Gamma\Rightarrow\Delta, @_i t^{@_i} = b}{@_i E^- t, \Gamma\Rightarrow\Delta}$$

$$(\Rightarrow E^-)\ \frac{\Gamma\Rightarrow\Delta, @_i Dt \quad @_i Ea, @_i t^{@_i} = a, \Gamma\Rightarrow\Delta}{\Gamma\Rightarrow\Delta, @_i E^- t}$$

$$(\approx\Rightarrow)\ \frac{@_i Dt_1, @_i Dt_2, @_i t_1^{@_i} = t_2^{@_i}, \Gamma\Rightarrow\Delta}{@_i t_1 \approx t_2, \Gamma\Rightarrow\Delta}$$

$$(\Rightarrow\approx)\ \frac{\Gamma\Rightarrow\Delta, @_i Dt_1 \quad \Gamma\Rightarrow\Delta, @_i Dt_2 \quad \Gamma\Rightarrow\Delta, @_i t_1^{@_i} = t_2^{@_i}}{\Gamma\Rightarrow\Delta, @_i t_1 \approx t_2}$$

A few conventions were applied in the schemata which will also be used in later sections. a denotes a parameter which is fresh, i.e. occurs only in displayed position and b is any parameter. $t^{@_i} := t$, if t is already rigid; otherwise for $t := c$ it is just $@_i c$ and for $t := f^n t_1, ..., t_n$ it is $@_i f^n t_1^{@_i}, ..., t_n^{@_i}$. Moreover, all rules for E and E^- are restricted to $t \notin PAR$. The notion of atomic formula covers formulae of the form $@_i P(r_1...r_n)$ and includes also cases where P is $=, D, E, E^-$; however, in the last two cases with restriction that $r \in PAR$.

The definition of proof is standard, as well as the notions of principal, side and parametric (context) formulae. Also the notion of the height of a proof is standard (the number of nodes of the longest branch). It is important to note that except (Cut) all rules satisfy the generalised subformula property to the effect that in premisses we have only subformulae of the conclusion closed for addition of sat-operator and the following kind of formulae: $@_i\Diamond j, @_i j, @_i Et, @_i Dt, @_i t = t'$. Moreover one can easily check that arguments of the last three atoms are either (rigidified) terms occuring in the conclusion or fresh parameters. This shows that from the proof-search perspective cut-free HFM is sufficiently analytic. In fact (FM) has also a disadvantage of putting every case in a form of cut-like rule with many premisses composed from formulae of the form $@_i\Diamond j, @_i j$. However, to concrete cases we can apply the rule-generation theorem from [17] which allows us to provide equivalent rules with lower branching factor and active formulae in the conclusion. Since frame expressivity is not our subject here we skip a discussion of these matters.

For cut elimination the key point is that all rules are substitutive and reductive. These notions were introduced by Ciabattoni [7] and applied for general form of cut elimination proof in hypersequent calculi by Metcalfe, Olivetti and Gabbay [23] but can be also applied in the present setting. The former property is connected with the fact that multisets of formulae may be safely substituted for a cut formula which is parametric. It allows for induction on the height of a proof in cases when the cut formula is not principal in at least one premiss of cut. Rules with side conditions concerning fresh parameters or nominals are not fully substitutive but due to the substitution lemma (see Appendix) this problem may be easily overcome. The latter property may be roughly defined as follows: A pair of introduction rules $(\Rightarrow \star)$, $(\star \Rightarrow)$ for a constant $\star$ is reductive if an application of cut on cut formulae introduced by these rules may be replaced by the series of cuts made on less complex formulae. Reductivity permits induction on cut-degree in the course of proving cut elimination. Of course the complexity c of all terms and formulae must be suitably defined:

$c(\alpha) = 0$ for $\alpha \in NOM \cup PAR \cup VAR$; $\quad c(t^{@_i}) = 1$;
$c(@_i\varphi) = c(\neg\varphi) = c(\varphi) + 1$; $c(Ox\varphi) = c(\varphi) + 2$, for $O \in \{\forall, \exists, \lambda\}$;
$c(\varphi \star \psi) = c(\varphi) + c(\psi) + 1$, for $\star$ being a binary connective;
$c(P(r_1, .., r_n)) = max\{c(r_1), ..., c(r_n)\} + 1$ (this clause includes $=$);
$c(Et) = c(E^-t) = c(t) + 2$; $c(Dt) = c(t) + 1$; $c(t_1 \approx t_2) = c(t_1) + c(t_2) + 1$;

For technical reasons we must assume that existence formulae have higher complexity than other atoms with the same arguments, and other rigid terms higher than parameters. One can check by inspection that all rules for compund formulae (including $\approx$) are reductive in pairs. As for $(Nom_1), (Nom_2), (= +)$ introducing nominal and ordinary equalities, they can be principal only in the right premiss of cut; in the left premiss they are always parametric formulae. The same remark applies to Dt which is principal in $(\approx -)$ but can also be principal in $(= +)$. In case of Et and E^-t two situations are possible. With $t \in PAR$ they may be principal also in the right premiss of cut due to $(= +)$, but in the left premiss only parametric. Otherwise in both premisses such formulae are principal due to specific rules for E and E^- which are reductive.

One may easily check that all rules in group 1-4 and 6 are validity-preserving in K models which implies soundness of HFM. The addition of specific rules generated by (FM) extends this to the class of all modal logics axiomatised by geometric formulae. Completeness proof for essentially equivalent propositional SC is provided by Bolander and Braüner [5] (see also [6]). Blackburn and ten Cate [4] provided completeness proof for axiomatic version of HFOML. Fitting and Mendelsohn [10] contain completeness proofs for tableaux formalization of group 6. In what follows instead of proving semantic completeness of HFM we apply a strategy similar to those utilized by Seligman [27] or Blackburn and Marx [3]. We will show that suitably restricted part of HFM is equivalent to FM tableau system by means of a series of purely syntactic transformations.

4 Auxiliary Systems and Transformations

Let us recall briefly FM tableaux (shortly FM-T) for the K-modality. For easier transformations and comparison with SC we will present it in Hintikka-style form, i.e. with sets of (prefixed) formulae as nodes of proof-trees, instead of the original Smullyan-style format defined on single formulae. Prefixes, denoting states in models, are finite lists of integers; for every prefix σ its one-digit extension $\sigma.n$ denotes a state which is accessible from σ (i.e. a state it denotes). Hence FM-T is a kind of external system with a special feature that prefixes are attached not only as prefixes to formulae but also, as subscripts, to terms (including parameters but not bound variables). t_σ syntactically encodes $\mathcal{I}_w(t)$ (where σ denotes w) and is always rigid. t^σ is t_σ if t is a name or a parameter, or just t if it is already rigid (i.e. a term where all components have subscripts). For complex terms, t^σ denotes the result of addition of σ as a subscript to all terms which are not subscripted so far. Rules for connectives are standard so we state only those for $\Box, \forall, =$ and the λ-operator:

$$(\Box)\ \frac{\sigma{:}\Box\varphi, \Gamma}{\sigma.n{:}\varphi, \sigma{:}\Box\varphi, \Gamma} \quad (\neg\Box)\ \frac{\sigma{:}\neg\Box\varphi, \Gamma}{\sigma.n{:}\neg\varphi, \Gamma} \quad (\lambda)\ \frac{\sigma{:}(\lambda x\varphi)t, \Gamma}{\sigma{:}\varphi[x/t^\sigma], \Gamma} \quad (\neg\lambda)\ \frac{\sigma{:}\neg(\lambda x\varphi)t, \Gamma}{\sigma{:}\neg\varphi[x/t^\sigma], \Gamma}$$

$$(\forall)\ \frac{\sigma{:}\forall x\varphi, \Gamma}{\sigma{:}\varphi[x/b_\sigma], \sigma{:}\forall x\varphi, \Gamma} \qquad (\neg\forall)\ \frac{\sigma{:}\neg\forall x\varphi, \Gamma}{\sigma{:}\neg\varphi[x/a_\sigma], \Gamma}$$

$$(=1)\ \frac{\Gamma}{\sigma{:}r = r, \Gamma} \qquad (=2)\ \frac{\sigma{:}r_1 = r_2, \sigma'{:}\varphi[x/r_1], \Gamma}{\sigma{:}r_1 = r_2, \sigma'{:}\varphi[x/r_1], \sigma'{:}\varphi[x/r_2], \Gamma}$$

Side conditions: $(\Box)$ $\sigma.n$ occurs in Γ; $(\neg\Box)$ $\sigma.n$ is fresh;
$(\forall)$ b_σ is any parameter with σ (i.e. occuring in Γ, otherwise a new one);
$(\neg\forall)$ a_σ is fresh; $(\neg\lambda)$ t^σ is defined; $(=2)$ φ any formula;
$(=1)$ σ occurs in Γ and r is defined (either a parameter or occurs in Γ);

We can get rid of prefixes and use instead the hybrid machinery of nominals with sat-operators obtaining a system HFM-T. It is enough to define a bijective mapping θ from prefixes to nominals in every FM-T proof. We systematically replace all occurences of prefixes in every node with a suitable $@_i$. Moreover, if $\sigma' := \sigma.n$, then we add $@_i\Diamond j$, where $@_i$ and $@_j$ correspond to σ and σ' respectively. As a result, for modals the new rules are obtained of the form:

$$(\Box')\ \frac{@_i\Box\varphi, @_i\Diamond j, \Gamma}{@_j\varphi, @_i\Box\varphi, @_i\Diamond j, \Gamma} \qquad (\neg\Box')\ \frac{@_i\neg\Box\varphi, \Gamma}{@_j\neg\varphi, @_i\Diamond j, \Gamma} \text{ with } j \text{ fresh}$$

This way every node Γ in FM-T proof is transformed into $\Delta, \theta\Gamma$, where $\theta\Gamma$ is a hybrid translation of Γ and Δ is the set of all $@_i\Diamond j$ such that $\sigma, \sigma.n$ occur in Γ, $\theta(\sigma) = i$ and $\theta(\sigma.n) = j$. Since θ can be converted, proofs in both systems are isomorphic and we obtain:

Lemma 4.1 $1 : \neg\varphi$ *has a closed tableau in FM-T iff* $@_i\neg\varphi$ *(for* i *not occurring in* φ*) has a closed tableau in HFM-T*

HFM-T differs from FM-T only in being an internal system. Since we want to work with SC we make two additional, rather cosmetic, changes. Sets are transformed into sequents by moving all negated formulae to succedents (with simultaneous deletion of negations) and finally turning all rules upside-down.

This way we obtain an auxiliary system HFM1 which is quite similar to HFM but for the time being in the language without $D, E, E^{-}, \approx$. On the other hand, in HFM1 also parameters are prefixed with $@_i$; we call them n-parameters. Moreover, we still need several side conditions, in particular we say that $@_i t$ is defined if t is a parameter or if it appears in the conclusion-sequent. In contrast to HFM, axioms are of the form $\Gamma \Rightarrow \Delta$ with $\Gamma \cap \Delta \neq \varnothing$, and structural rules from group 1 are not required. Propositional part (group 2, 3) is like in HFM but with two slightly different modal rules:

$$(\Box \Rightarrow') \ \frac{@_i\Box\varphi, @_j\varphi, @_i\Diamond j, \Gamma \Rightarrow \Delta}{@_i\Box\varphi, @_i\Diamond j, \Gamma \Rightarrow \Delta} \qquad (\Rightarrow\Diamond') \ \frac{@_i\Diamond j, \Gamma \Rightarrow \Delta, @_i\Diamond\varphi, @_j\varphi}{@_i\Diamond j, \Gamma \Rightarrow \Delta, @_i\Diamond\varphi}$$

The remaining rules look like that:

$$(\forall\Rightarrow) \ \frac{@_i\varphi[x/@_i b], @_i\forall x\varphi, \Gamma \Rightarrow \Delta}{@_i\forall x\varphi, \Gamma \Rightarrow \Delta} \qquad (\Rightarrow\forall) \ \frac{\Gamma \Rightarrow \Delta, @_i\varphi[x/@_i a]}{\Gamma \Rightarrow \Delta, @_i\forall x\varphi}$$

$$(\exists \Rightarrow) \ \frac{@_i\varphi[x/@_i a], \Gamma \Rightarrow \Delta}{@_i\exists x\varphi, \Gamma \Rightarrow \Delta} \qquad (\Rightarrow\exists) \ \frac{\Gamma \Rightarrow \Delta, @_i\exists x\varphi, @_i\varphi[x/@_i b]}{\Gamma \Rightarrow \Delta, @_i\exists x\varphi}$$

$$(= -) \ \frac{@_i r = r, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta} \qquad (= +') \ \frac{@_i\varphi[x/r_1], @_j r_1 = r_2, @_i\varphi[x/r_2], \Gamma \Rightarrow \Delta}{@_j r_1 = r_2, @_i\varphi[x/r_2], \Gamma \Rightarrow \Delta}$$

$$(\lambda\Rightarrow') \ \frac{@_i\varphi[x/t^{@_i}], \Gamma \Rightarrow \Delta}{@_i(\lambda x\varphi)t, \Gamma \Rightarrow \Delta} \qquad (\Rightarrow\lambda') \ \frac{\Gamma \Rightarrow \Delta, @_i\varphi[x/t^{@_i}]}{\Gamma \Rightarrow \Delta, @_i(\lambda x\varphi)t}$$

where: $@_i a$ is fresh, whereas $@_i b$ occurs in Γ, Δ, otherwise it is also fresh; r in $(= -)$ and $t^{@_i}$ in $(\Rightarrow \lambda')$ are defined; φ in $(= +')$ is any formula.

Lemma 4.2 *$@_i\neg\varphi$ has closed tableau in HFM-T iff HFM1* $\vdash\ \Rightarrow @_i\varphi$

The above lemma holds trivially. HFM1 is a quite well-behaved system, however the application of sat-operators to parameters seems to be excessive. After all, parameters are rigid by definition and do not need special indication for that. The problem is that in FM-T the addition of prefixes to parameters plays an additional function; it indicates an existence in a state denoted by a prefix. This of course was transmitted also to HFM1. However, this function can be performed by using the existence predicate, as in the case of free logics. The situation is the same with definedness. In HFM1, exactly as in FM-T, this information is carried out by side conditions added to some rules. Again it is possible to make it explicit by introduction of definedness predicate. In fact, all this, and even more, is present in FM-T [10] in the form of definitions introduced for more compact expression of interesting features of this system:

$$Dt := (\lambda x x = x)t \qquad Et := (\lambda x \exists y(y = x))t$$
$$E^{-}t := (\lambda x \neg\exists y(y = x))t \qquad t_1 \approx t_2 := (\lambda x(\lambda y x = y)t_2)t_1$$

From this list only the first two are necessary for obtaining the results mentioned above on explicit representation of existence and definedness by means of special formulae instead of side conditions. However, the nonexistence predicate E^- (originally $\bar{E}$) is important for showing the real difference between existence and definedness which is expressed by the thesis $Dt \leftrightarrow Et \vee E^-t$. Term equality $\approx$ is important for definition of suitable rules for definite descriptions in section 5, so we introduce it as well. HFM1 can be enriched in a similar way as FM-T, by introducing definitions for lacking constant predicates. However, it is better to add them as eight additional axioms. For example, for D they have the form:

$$@_iDt, \Gamma \Rightarrow \Delta, @_i(\lambda xx = x)t \quad \text{and} \quad @_i(\lambda xx = x)t, \Gamma \Rightarrow \Delta, @_iDt$$

Moreover, let us notice that we can add to HFM1 three admissible rules: two rules of weakening and cut. Admissibility of weakening can be easilly proven syntactically but with cut it is not so simple. However, cut is validity-preserving and HFM1 is complete, hence admissibility of cut in HFM1 follows. Now we can define another system HFM2 which is based on the same language as HFM, i.e. with $D, E, E^-, \approx$ primitive and without sat-operators attached to parameters. In the propositional basis it is exactly as HFM1, including modal rules; $(= +')$ is also the same. The remaining rules are closer to HFM; in particular $(\Rightarrow \forall E), (\exists E \Rightarrow), (\lambda \Rightarrow)$ are the same, the other ones are:

$$(\forall E{\Rightarrow}') \quad \frac{@_i\varphi[x/b], @_i\forall x\varphi, @_iEb, \Gamma \Rightarrow \Delta}{@_i\forall x\varphi, @_iEb, \Gamma \Rightarrow \Delta} \qquad ({\Rightarrow}\lambda'') \quad \frac{@_iDt, \Gamma \Rightarrow \Delta, @_i\varphi[x/t^{@_i}]}{@_iDt, \Gamma \Rightarrow \Delta, @_i(\lambda x\varphi)t}$$

$$({\Rightarrow}\exists E') \quad \frac{@_iEb, \Gamma \Rightarrow \Delta, @_i\exists x\varphi, @_i\varphi[x/b]}{@_iEb, \Gamma \Rightarrow \Delta, @_i\exists x\varphi} \qquad (\approx -') \quad \frac{@_iDt, @_it \approx t, \Gamma \Rightarrow \Delta}{@_iDt, \ \Gamma \Rightarrow \Delta}$$

Note that $(= -)$ is replaced with $(\approx -')$. We add $(E-), (D-)$ as in HFM; these rules are necessary to make explicit what was implicit in HFM1. $(E-)$ is required since in FM semantics world-domains are nonempty. In HFM1 the side condition for $(\forall \Rightarrow)$ and $(\Rightarrow \exists)$ permits introduction of a n-parameter even if no previous application of $(\Rightarrow \forall)$ or $(\exists \Rightarrow)$ provided some. For the present rules an existence formula must be already present in the antecedent, so in case there is no such formula we apply $(E-)$ first (in root first proof-search). Note that if we drop this rule we obtain a variant for logics admitting empty domains. $(D-)$ explicitly shows that (all) parameters are defined. Definedness formulae are also introduced to $(\Rightarrow \lambda'')$ and $(\approx -')$, whereas such a formula added in the premiss-sequent of $(\lambda \Rightarrow)$ plays a similar role for names as in $(D-)$ for parameters. There is one small difference with the original conditions stated by Fitting and Mendelsohn for rules demanding already defined terms. They require that suitable rigid terms should be defined hence if we follow strictly their formulation our definedness formulae in both rules for λ and $(\approx -')$ should be of the form $@_iDt^{@_i}$ instead of $@_iDt$. However, one can easily check that $@_iDt^{@_i}$ and $@_iDt$ are semantically equivalent and for technical reasons using $@_iDt$ in rules is simpler. Instead of definitional axioms we add to HFM2 suitable introduction rules for all additional predicates except D. Rules $(E \Rightarrow)$, $(E^- \Rightarrow)$ and $(\approx\Rightarrow)$ are the same as in HFM; the remaining ones are:

$$(\Rightarrow E')\ \ \frac{@_i Dt, \Gamma \Rightarrow \Delta, @_i Eb \quad @_i Dt, \Gamma \Rightarrow \Delta, @_i t^{@_i} = b}{@_i Dt, \Gamma \Rightarrow \Delta, @_i Et}$$

$$(\Rightarrow E^-)\ \ \frac{@_i Dt, @_i Ea, @_i t^{@_i} = a, \Gamma \Rightarrow \Delta}{@_i Dt, \Gamma \Rightarrow \Delta, @_i E^- t}$$

$$(\Rightarrow \approx')\ \ \frac{@_i Dt_1, @_i Dt_2, \Gamma \Rightarrow \Delta, @_i t_1^{@_i} = t_2^{@_i}}{@_i Dt_1, @_i Dt_2, \Gamma \Rightarrow \Delta, @_i t_1 \approx t_2}$$

As in HFM, in rules for E, E^-, $t \notin PAR$. For D no special rules are needed. Despite several differences concerning rules, and using n-parameters in HFM1 and parameters in HFM2, both systems are equivalent in the sense of provability of the same sequents containing sentences (see Appendix).

Now we are ready to compare HFM2 with HFM. It is straightforward to prove that all rules of HFM2, with the exception of $(= +')$, are derivable in HFM; it is sufficient to apply rule-generation theorem from [17]. $(= +')$ is derivable additionally by induction on the complexity of φ with $(= +)$ in the basis. We cannot in general prove the opposite since already the background propositional hybrid part of HFM contains elements not expressible in FM, like nominal rules (group 4) or rules expressing frame conditions (group 5). This is basically a difference between the expressive powers of external labelled system, as exemplified here by prefixed tableau calculus of Fitting and Mendelsohn, and internal labelled system; the latter have much greater expressive power. However, for the part of HFM restricted to rules from group 1-3 and 6, again rule-generation theorem suffices to demostrate their derivability in HFM2. Therefore HFM2 and HFM (restricted to rules 1-3, 6) are equivalent (see Appendix).

5 Definite Descriptions

We postponed a treatment of definite descriptions since it requires some additional changes. In particular, categories of formulae FOR and terms $TERM$ must be defined simultaneously, and similarly the notion of interpretation of terms and satisfaction of formulae must be treated together. On the other hand, all terms considered so far can be represented as definite descriptions hence we can reduce the category of terms accordingly. We must add the iota-operator $\imath$, for forming definite descriptions out of formulae:

- if $\varphi \in FOR$, then $\imath x\varphi \in TERM$.

Semantically we characterise it by the following clause:

- $\mathcal{I}^v_w(\imath x\varphi) = o$ iff $\mathfrak{M}, v^x_o, w \vDash \varphi$ and no other x-variant of v satisfies φ in w.

Hence definite descriptions are also non-rigid and may be undefined in some (possibly all) worlds. Again addition of $@_i$ to definite description makes it a rigid term; a name of its designatum in $\mathcal{I}(i)$, if it is defined there. Complexity $c(\imath x\varphi) = c(\varphi) + 1$ but for $c(@_i\imath x\varphi) = 1$ which is fixed for all rigid terms.

Syntactically, Fitting and Mendelsohn's approach is based on the form of Hintikka Axiom, of which a "tentative version" takes the form:

$H: \ t \approx \imath x\varphi \leftrightarrow (\lambda x\varphi)t \wedge \forall y(\varphi[x/y] \to (\lambda xx = y)t)$, where y is not in φ;

which corresponds directly to the semantic clause. Note however that universal quantifier in H is possibilistic, i.e. it ranges over all elements of the (frame) domain. In FM tableaux its essential content is represented by means of three rules of introduction of implications corresponding to both directions of H. They are weaker in the sense that universal quantifier which introduces unwanted existential commitments is eliminated, and the rule corresponding to $H^{\leftarrow}$, i.e. to the right-left implication, introduces not valid but satisfiable formula containing labelled parameter. Such rules cannot be directly transformed into well-behaved SC rules so, instead of dealing with three FM rules, we introduce three other ones below and directly show that: (1) Hintikka axiom H, as restated in [10], is provable in HFM with these rules; (2) three additional $\imath$-rules are derivable in HFM in the presence of sequent $\Rightarrow H$ as an additional axiom. To realise this aim we must add to our language additional, possibilistic quantifiers symbolised by Tarskian $\bigwedge, \bigvee$. Semantically they are characterised:

- $\mathfrak{M}, v, w \vDash \bigwedge x\varphi$ iff $\mathfrak{M}, v^x_o, w \vDash \varphi$ for all $o \in \mathcal{D}$
- $\mathfrak{M}, v, w \vDash \bigvee x\varphi$ iff $\mathfrak{M}, v^x_o, w \vDash \varphi$ for some $o \in \mathcal{D}$

Now the system HFMD is HFM with the following additional rules:

$$(\bigwedge\Rightarrow)\ \frac{\Gamma\Rightarrow \Delta, @_iDt \quad @_i\varphi[x/t^{@_i}], \Gamma\Rightarrow \Delta}{@_i \bigwedge x\varphi, \Gamma\Rightarrow \Delta} \qquad (\Rightarrow\bigwedge)\ \frac{\Gamma\Rightarrow \Delta, @_i\varphi[x/a]}{\Gamma\Rightarrow \Delta, @_i \bigwedge x\varphi}$$

$$(\Rightarrow\bigvee)\ \frac{\Gamma\Rightarrow \Delta, @_iDt \quad \Gamma\Rightarrow \Delta, @_i\varphi[x/t^{@_i}]}{\Gamma\Rightarrow \Delta, @_i \bigvee x\varphi} \qquad (\bigvee\Rightarrow)\ \frac{@_i\varphi[x/a], \Gamma\Rightarrow \Delta}{@_i \bigvee x\varphi, \Gamma\Rightarrow \Delta}$$

$$(\imath\Rightarrow 1)\ \frac{@_iDt, @_i\varphi[x/t^{@_i}], \Gamma\Rightarrow \Delta}{@_it \approx \imath x\varphi, \Gamma\Rightarrow \Delta}$$

$$(\imath\Rightarrow 2)\ \frac{\Gamma\Rightarrow \Delta, @_i\varphi[x/b] \quad @_iDt, @_ib = t^{@_i}, \Gamma\Rightarrow \Delta}{@_it \approx \imath x\varphi, \Gamma\Rightarrow \Delta}$$

$$(\Rightarrow\imath)\ \frac{\Gamma\Rightarrow \Delta, @_iDt \quad \Gamma\Rightarrow \Delta, @_i\varphi[x/t^{@_i}] \quad @_i\varphi[x/a], \Gamma\Rightarrow \Delta, @_ia = t^{@_i}}{\Gamma\Rightarrow \Delta, @_it \approx \imath x\varphi}$$

and with $(\approx\Rightarrow), (\Rightarrow\approx)$ replaced with:

$$(\approx r1)\ \frac{\Gamma\Rightarrow \Delta, @_it_1 \approx t_2 \quad @_it_1^{@_i} = t_2^{@_i}, \Gamma\Rightarrow \Delta}{\Gamma\Rightarrow \Delta}$$

$$(\approx r2)\ \frac{\Gamma\Rightarrow \Delta, @_it_1 \approx t_2 \quad @_iDt_i, \Gamma\Rightarrow \Delta}{\Gamma\Rightarrow \Delta}$$

$$(r\approx)\ \frac{\Gamma\Rightarrow \Delta, @_it_1^{@_i} = t_2^{@_i} \quad @_it_1 \approx t_2, \Gamma\Rightarrow \Delta}{@_iDt_1, @_iDt_2, \Gamma\Rightarrow \Delta}$$

HFMD is adequate since we can prove that rules for $\imath$ are interderivable with $\Rightarrow H$, moreover, cut elimination also holds for it (see Appendix) although this requires some comment. First note that in $(\bigwedge \Rightarrow), (\Rightarrow \bigvee)$ we instantiate variable x with any rigid term but since they have complexity 1, the new rules are also reductive and the proof of cut elimination for HFM is not spoiled. We cannot use restricted form of instantiation, like in case of $(\forall E \Rightarrow), (\Rightarrow \exists E)$, since such system would be incomplete. Consider two formulae: $Er \wedge \forall x\varphi \rightarrow \varphi[x/r]$ and $Dr \wedge \bigwedge x\varphi \rightarrow \varphi[x/r]$. Both are valid and the former is provable in HFM but the second would be unprovable if $(\bigwedge \Rightarrow)$ is restricted to parameters. On the other hand in $(\Rightarrow \bigwedge), (\bigvee \Rightarrow)$ we do not need the additional formula $@_i Da$ in the premiss since parameters are defined everywhere.

Note also that rules of HFMD are defined in such a way that the situation is excluded where some cut formula is principal in both premisses of cut but obtained by means of different kind of rules which are not reductive. In particular, term equality is introduced only by means of rules for $\imath$ and they are reductive. It is the reason why in HFMD we have to replace rules for $\approx$ from HFM with apparently worse equivalents. Since principal formulae of rules for definite descriptions are term equalities there would be a clash with HFM rules. Consider a situation when both cut formulae are term equalities but one is introduced by means of an $\imath$-rule and the other by means of $(\approx\Rightarrow)$ or $(\Rightarrow\approx)$ – an induction on cut-degree fails. However, such situation cannot happen in HFMD where instead of $(\approx\Rightarrow)$ and $(\Rightarrow\approx)$ we have $(\approx r1), (\approx r2), (r \approx)$ which are safe in this respect. Clearly, the new rules are equivalent to $(\approx\Rightarrow), (\Rightarrow\approx)$ (by rule-generation theorem [17] mentioned in section 3) although worse from the proof-search standpoint.

6 Conclusion

We have shown that HL is a sufficiently flexible framework for expressing FM version of FOML. Moreover, in this setting we can formulate a well-behaved SC admitting cut elimination. Although we did not provide a semantic completeness proof it can be carried out either by using a strategy from [4] which is more standard and requires cut, or by means of Hintikka-style saturation technique like in [3] which is possible due to proved cut elimination. Moreover, HFM is formulated in the weak hybrid language as a basis; all additions were taken from FOML of Fitting and Mendelsohn. We can still enrich the language with specifically hybrid constants like nominal quantifiers or $\downarrow$-operator.

On the other hand, our treatment of definite descriptions by means of rules using $\approx$ may be seen as not wholly satisfactory. We obtain a system where $=$ is better characterised than in Indrzejczak [16] but at the cost of some redundancy – two kinds of equality are applied that differ only syntactically but not semantically. The other option would be to characterize definite descriptions by means of special rules for definedness formulae. This is the approach explored by Orlandelli [24] in the framework of labelled SC. In his system definedness predicate is not a part of a language but rather a technical device of the shape $D(t, x, w)$ meaning that t denotes x in w. In our approach this information is divided between $\approx$ and D as unary predicate. The lack of space forbids more extensive comparison of both approaches.

We should add that the treatment of definite descriptions either in terms of rules using some kind of equality or a predicate of definedness still does not provide the characteristics which is separate, in the sense of not exposing other constants in rules except $\imath$-operator. It is worth to explore more general perspective which is connected with using terms on a par with formulae in sequents. This device, introduced by Jaśkowski [20] in his first formulation of ND, was recently succesfully applied in many contexts, for example by Textor [28], Restall [26], Gazzari [12] and Indrzejczak [18]. In hybrid languages a uniform application of sat-operators to formulae and terms seems to offer a particularly interesting and uniform perspective where items are just sat-phrases be it either a formula or a term. We leave this problem for further study.

Appendix

Lemma .1 *If HFM1* $\vdash\ \Gamma \Rightarrow \Delta$*, then HFM2* $\vdash\ \Gamma \Rightarrow \Delta$*, where* Γ, Δ *contain only sentences.*

Proof: Consider a HFM1-proof $\mathcal{D}$. Let $\{p_1, \ldots p_n\}$ be the set of all distinct n-parameters occuring in a proof and $\{p'_1, \ldots p'_n\}$ the corresponding set of distinct parameters. Going from the root to leaves define for every node $\Gamma \Rightarrow \Delta$ a corresponding sequent $\Pi, \Gamma' \Rightarrow \Delta'$ where Γ', Δ' are obtained from Γ, Δ by simultaneous substitution of every n-parameter p_i with corresponding parameter p'_i. Moreover let $\Pi = \{@_k Ep'_i : p_i := @_k p'_i \in PAR(\Gamma \cup \Delta)\} \cup \{@_i Dt : @_i t \text{ is defined in } \Gamma \cup \Delta\}$. This way we obtain an isomorphic tree $\mathcal{D}'$ of sequents with the same root in the language of HFM2. This tree is not necessarily a HFM2-proof so we must systematically made some adjustments. All leaves of $\mathcal{D}$ which are instances of Ax are trivially axioms of HFM2. However, for the cases of eight definitional axioms we have to provide proofs of their corresponding sequents in HFM2. For two axioms characterising D we have:

$$
\cfrac{\cfrac{\cfrac{@_i Dt, \Pi, @_i(@_i t = @_i t), \Gamma \Rightarrow \Delta, @_i(@_i t = @_i t)}{@_i Dt, \Pi, @_i(@_i t = @_i t), \Gamma, \Rightarrow \Delta, @_i(\lambda x x = x)t}\ (\Rightarrow \lambda'')}{@_i Dt, \Pi, @_i(t \approx t), \Gamma, \Rightarrow \Delta, @_i(\lambda x x = x)t}\ (\approx\Rightarrow)}{@_i Dt, \Pi, \Gamma \Rightarrow \Delta, @_i(\lambda x x = x)t}\ (\approx -')
$$

and

$$
(\lambda \Rightarrow)\ \frac{@_i Dt, \Pi, @_i(@_i t = @_i t), \Gamma \Rightarrow \Delta, @_i Dt}{\Pi, @_i(\lambda x x = x)t, \Gamma \Rightarrow \Delta, @_i Dt}
$$

For E with $t \notin PAR$:

$$
\cfrac{\cfrac{\cfrac{@_i Dt, \Pi, @_i Ea, @_i(a = @_i t), \Gamma \Rightarrow \Delta, @_i(a = @_i t)}{@_i Dt, \Pi, @_i Ea, @_i(a = @_i t), \Gamma \Rightarrow \Delta, @_i \exists y(y = @_i t)}\ (\Rightarrow \exists E')}{@_i Dt, \Pi, @_i Ea, @_i(a = @_i t), \Gamma \Rightarrow \Delta, @_i(\lambda x \exists y y = x)t}\ (\Rightarrow \lambda'')}{\Pi, @_i Et, \Gamma \Rightarrow \Delta, @_i(\lambda x \exists y y = x)t}\ (E \Rightarrow)
$$

$$
(\lambda\Rightarrow)\,\dfrac{(\exists E\Rightarrow)\,\dfrac{(\Rightarrow E')\,\dfrac{\Sigma,\Pi,\Gamma\Rightarrow\Delta,@_iEa \qquad \Sigma,\Pi,\Gamma\Rightarrow\Delta,@_i(a=@_it)}{@_iDt,@_iEa,\Pi,@_i(a=@_it),\Gamma\Rightarrow\Delta,@_iEt}}{@_iDt,\Pi,@_i\exists y(y=@_it),\Gamma\Rightarrow\Delta,@_iEt}}{\Pi,@_i(\lambda x\exists yy=x)t,\Gamma\Rightarrow\Delta,@_iEt}
$$

where $\Sigma=\{@_iDt,@_iEa,@_i(a=@_it)\}$. If $t\in PAR$ both sequents are provable without $(E\Rightarrow),(\Rightarrow E')$; it justifies our restriction on their application.

For all applications of propositional rules in $\mathcal{D}$ we do not need any changes in $\mathcal{D}'$. For $(\forall\Rightarrow)$ we have by definition of Π that $@_iEb\in\Pi$ and that $@_i\varphi[x/b]\in\Gamma'$, so in case $@_ib$ was already present in the conclusion of the application of this rule in $\mathcal{D}$ we need no change in $\mathcal{D}'$. In case $@_ib$ was fresh in this application of $(\forall\Rightarrow)$ we must add $@_iEb$ to Π in the conclusion to secure the correctness of $(\forall E\Rightarrow)$ in HFM2 and apply next $(E-)$ to remove $@_iEb$. For $(\Rightarrow\forall)$ $@_ia$ occurs only in $@_i\varphi$ but in the corresponding sequent of $\mathcal{D}'$ a occurs also in $@_iDa\in\Pi$, in $@_iEa\in\Pi$ and in $@_i\varphi\in\Delta'$. Therefore first we delete $@_iDa$ by application of $(D-)$, then the application of $(\Rightarrow\forall E)$ on resulting sequent is correct and yields the desired conclusion. Applications of $(=+'),(\lambda\Rightarrow')$ and $(\Rightarrow\lambda')$ in $\mathcal{D}$ correspond to correct applications of HFM2-versions of respective rules in $\mathcal{D}'$ by definition of Π. For $(=-)$ we apply $(\approx-')$ and $(\approx\Rightarrow)$. $\square$

Before proving the converse let us first demonstrate a derivability of all specific rules of HFM2 in HFM1. Clearly instead of parameters we will use n-parameters here. Quantifier rules $(\forall E\Rightarrow')$ and $(\Rightarrow\exists E')$ are just special versions of HFM1 rules. Derivability of $(\Rightarrow\forall E)$ and $(\exists E\Rightarrow)$ needs a demonstration:

$$
(\Rightarrow\forall)\,\dfrac{(=-)\,\dfrac{(Cut)\,\dfrac{(Cut)\,\dfrac{\mathcal{D} \qquad @_i(\lambda x\exists yy=x)@_ia\Rightarrow @_iE@_ia}{@_i(@_ia=@_ia)\Rightarrow @_iE@_ia} \qquad @_iE@_ia,\Gamma\Rightarrow\Delta,@_i\varphi[x/@_ia]}{@_i(@_ia=@_ia),\Gamma\Rightarrow\Delta,@_i\varphi[x/@_ia]}}{\Gamma\Rightarrow\Delta,@_i\varphi[x/@_ia]}}{\Gamma\Rightarrow\Delta,@_i\forall x\varphi}
$$

where $\mathcal{D}$ is:

$$
(\Rightarrow\lambda')\,\dfrac{(\Rightarrow\exists)\,\dfrac{@_i(@_ia=@_ia)\Rightarrow @_i\exists yy=@_ia,@_i(@_ia=@_ia)}{@_i(@_ia=@_ia)\Rightarrow @_i\exists yy=@_ia}}{@_i(@_ia=@_ia)\Rightarrow @_i(\lambda x\exists yy=x)@_ia}
$$

Similarly for $(\exists E\Rightarrow)$. $(\Rightarrow\lambda'')$ needs no justification; for $(\lambda\Rightarrow)$ we have:

$$
(\lambda\Rightarrow')\,\dfrac{(=-)\,\dfrac{(Cut)\,\dfrac{(Cut)\,\dfrac{(\Rightarrow\lambda')\,\dfrac{@_i(t^{@_i}=t^{@_i})\Rightarrow @_i(t^{@_i}=t^{@_i})}{@_i(t^{@_i}=t^{@_i})\Rightarrow @_i(\lambda xx=x)t} \qquad @_i(\lambda xx=x)t\Rightarrow @_iDt}{@_i(t^{@_i}=t^{@_i})\Rightarrow @_iDt} \qquad @_iDt,@_i\varphi[x/t^{@_i}],\Gamma\Rightarrow\Delta}{@_i(t^{@_i}=t^{@_i}),@_i\varphi[x/t^{@_i}],\Gamma\Rightarrow\Delta}}{@_i\varphi[x/t^{@_i}],\Gamma\Rightarrow\Delta}}{@_i(\lambda x\varphi)t,\Gamma\Rightarrow\Delta}
$$

Derivability of $(\approx-')$:

$$\dfrac{\mathcal{D} \qquad \dfrac{@_i(\lambda x(\lambda yy = x)t)t \Rightarrow @_i(t \approx t) \qquad @_i(t \approx t), @_iDt, \Gamma \Rightarrow \Delta}{@_i(\lambda x(\lambda yy = x)t)t, @_iDt, \Gamma \Rightarrow \Delta}\,(Cut)}{@_iDt, \Gamma \Rightarrow \Delta}\,(Cut)$$

where $\mathcal{D}$ is:

$$\dfrac{@_iDt \Rightarrow @_i(\lambda xx = x)t \qquad \dfrac{\dfrac{\dfrac{@_i(@_it = @_it) \Rightarrow @_i(@_it = @_it)}{@_i(@_it = @_it) \Rightarrow @_i(\lambda yy = @_it)t}\,(\Rightarrow \lambda')}{@_i(@_it = @_it) \Rightarrow @_i(\lambda x(\lambda yy = x)t)t}\,(\Rightarrow \lambda')}{@_i(\lambda xx = x)t \Rightarrow @_i(\lambda x(\lambda yy = x)t)t}\,(\lambda \Rightarrow')}{@_iDt \Rightarrow @_i(\lambda x(\lambda yy = x)t)t}\,(Cut)$$

Derivability of other rules goes similarly, hence we obtain:

Lemma .2 *If HFM2* $\vdash \ \Gamma \Rightarrow \Delta$*, then HFM1* $\vdash \ \Gamma \Rightarrow \Delta$

Proof by induction on the height of HFM2-proof. Clearly as a preliminary step we must provide a reverse substitution in all nodes of HFM2-proof, i.e. in Γ, Δ all different parameters must be substituted with different n-parameters in such a way that for any $@_iEb$ or $@_iDb$ in Γ, Δ, b is substituted with $@_ib$. Derivability of all specific rules of HFM2 in HFM1 suffices for the proof. □

Lemma .3 *Provability of H in HFMD*

$$(\Rightarrow \wedge)\,\dfrac{(\imath \Rightarrow 1)\,\dfrac{(\Rightarrow \lambda)\,\dfrac{@_iDt \Rightarrow @_iDt \qquad @_i\varphi[x/t^{@_i}] \Rightarrow @_i\varphi[x/t^{@_i}]}{@_iDt, @_i\varphi[x/t^{@_i}] \Rightarrow @_i(\lambda x\varphi)t}}{@_it \approx \imath x\varphi \Rightarrow @_i(\lambda x\varphi)t} \qquad \mathcal{D}}{@_it \approx \imath x\varphi \Rightarrow @_i(\lambda x\varphi)t \wedge \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t)}$$

where $\mathcal{D}$ is:

$$(\Rightarrow \bigwedge)\,\dfrac{(\Rightarrow\rightarrow)\,\dfrac{\dfrac{@_i\varphi[x/a] \Rightarrow @_i\varphi[x/a] \qquad \dfrac{@_iDt \Rightarrow @_iDt \qquad @_it^{@_i} = a \Rightarrow @_it^{@_i} = a}{@_iDt, @_it^{@_i} = a \Rightarrow @_i(\lambda xx = a)t}\,(\Rightarrow \lambda)}{@_it \approx \imath x\varphi, @_i\varphi[x/a] \Rightarrow @_i(\lambda xx = a)t}\,(\imath \Rightarrow 2)}{@_it \approx \imath x\varphi, \Rightarrow @_i\varphi[x/a] \rightarrow (\lambda xx = a)t}}{@_it \approx \imath x\varphi \Rightarrow @_i \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t)}$$

Next, the converse:

$$(\wedge \Rightarrow)\,\dfrac{(\lambda \Rightarrow)\,\dfrac{(\Rightarrow \imath)\,\dfrac{@_iDt \Rightarrow @_iDt \qquad @_i\varphi[x/t^{@_i}] \Rightarrow @_i\varphi[x/t^{@_i}] \qquad \mathcal{D}}{@_iDt, @_i\varphi[x/t^{@_i}], @_i \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t) \Rightarrow @_it \approx \imath x\varphi}}{@_i(\lambda x\varphi)t, @_i \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t) \Rightarrow @_it \approx \imath x\varphi}}{@_i(\lambda x\varphi)t \wedge \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t) \Rightarrow @_it \approx \imath x\varphi}$$

where $\mathcal{D}$ is:

$$\dfrac{(D-)\,\dfrac{@_iDa \Rightarrow @_iDa}{\Rightarrow @_iDa} \qquad \dfrac{@_i\varphi[x/a] \Rightarrow @_i\varphi[x/a] \qquad \dfrac{@_iDt, @_it^{@_i} = a \Rightarrow @_it^{@_i} = a}{@_i(\lambda xx = a)t \Rightarrow @_it^{@_i} = a}\,(\lambda \Rightarrow)}{@_i\varphi[x/a] \rightarrow (\lambda xx = a)t, @_i\varphi[x/a] \Rightarrow @_it^{@_i} = a}\,(\rightarrow\Rightarrow)}{@_i \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t), @_i\varphi[x/a] \Rightarrow @_it^{@_i} = a}\,(\bigwedge \Rightarrow)$$

Lemma .4 *Derivability of HFMD rules (the case of $(\imath \Rightarrow 1), (\imath \Rightarrow 2)$)*

$$\dfrac{\Rightarrow H^{\rightarrow} \qquad \dfrac{@_i t \approx \imath x\varphi \Rightarrow @_i t \approx \imath x\varphi \qquad \dfrac{\dfrac{\dfrac{@_i Dt, @_i\varphi[x/t^{@_i}], \Gamma \Rightarrow \Delta}{@_i(\lambda x\varphi)t, \Gamma \Rightarrow \Delta}\,(\lambda\Rightarrow)}{@_i(\lambda x\varphi)t, @_i \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t), \Gamma \Rightarrow \Delta}\,(W\Rightarrow)}{@_i(\lambda x\varphi)t \wedge \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t), \Gamma \Rightarrow \Delta}\,(\wedge\Rightarrow)}{@_i t \approx \imath x\varphi, @_i t \approx \imath x\varphi \rightarrow (\lambda x\varphi)t \wedge \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t), \Gamma \Rightarrow \Delta}\,(\rightarrow\Rightarrow)}{@_i t \approx \imath x\varphi, \Gamma \Rightarrow \Delta}\,(Cut)$$

$$\dfrac{\dfrac{\dfrac{(D-)\,\dfrac{@_i Db \Rightarrow @_i Db}{\Rightarrow @_i Db} \qquad \dfrac{\Gamma \Rightarrow \Delta, @_i\varphi[x/b] \qquad \dfrac{@_i Dt, @_i t^{@_i} = b, \Gamma \Rightarrow \Delta}{@_i(\lambda xx = b)t, \Gamma \Rightarrow \Delta}\,(\lambda\Rightarrow)}{@_i\varphi[x/b] \rightarrow (\lambda xx = b)t, \Gamma \Rightarrow \Delta}\,(\rightarrow\Rightarrow)}{@_i \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t), \Gamma \Rightarrow \Delta}\,(\bigwedge\Rightarrow)}{@_i(\lambda x\varphi)t, @_i \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t), \Gamma \Rightarrow \Delta}\,(W\Rightarrow)}{@_i(\lambda x\varphi)t \wedge \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t), \Gamma \Rightarrow \Delta}\,(\wedge\Rightarrow)$$

and obtain the conclusion of $(\imath \Rightarrow 2)$ by cut with:

$$\dfrac{\Rightarrow H^{\rightarrow} \qquad \dfrac{@_i t \approx \imath x\varphi \Rightarrow @_i t \approx \imath x\varphi \qquad @_i(\lambda x\varphi)t \wedge \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t) \Rightarrow \psi}{@_i t \approx \imath x\varphi, @_i t \approx \imath x\varphi \rightarrow (\lambda x\varphi)t \wedge \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t) \Rightarrow \psi}\,(\rightarrow\Rightarrow)}{@_i t \approx \imath x\varphi \Rightarrow @_i(\lambda x\varphi)t \wedge \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t)}\,(Cut)$$

where $\psi := @_i(\lambda x\varphi)t \wedge \bigwedge y(\varphi[x/y] \rightarrow (\lambda xx = y)t)$
We prove derivability of $(\Rightarrow \imath)$ in a similar way. □

To prove cut elimination first note that for HFM and HFMD holds:

Lemma .5 (Height-preserving Substitution)
If $\vdash_k \Gamma \Rightarrow \Delta$, then $\vdash_k (\Gamma \Rightarrow \Delta)[i/j]$;
If $\vdash_k \Gamma \Rightarrow \Delta$, then $\vdash_k (\Gamma \Rightarrow \Delta)[a/r]$.

By lemma 5 every proof may be systematically transformed into regular proof – every fresh parameter and nominal is fresh in the entire proof.

Let cut-degree of cut-formula $@_i\varphi$ be its complexity, i.e. $d@_i\varphi = c(@_i\varphi)$ and proof-degree ($d\mathcal{D}$) be the maximal cut-degree in $\mathcal{D}$.

Technically the proof of cut elimination theorem is an extension of the proof for propositional HL in Indrzejczak [15] (see also [23], [16]) and is based on:

Lemma .6 (Right reduction) *Let $\mathcal{D}_1 \vdash \Gamma \Rightarrow \Delta, @_i\varphi$ and $\mathcal{D}_2 \vdash @_i\varphi^n, \Pi \Rightarrow \Sigma$ and $d\mathcal{D}_1, d\mathcal{D}_2 < d@_i\varphi$, and $@_i\varphi$ principal in $\Gamma \Rightarrow \Delta, @_i\varphi$, then we can construct a proof $\mathcal{D}$ such that $\mathcal{D} \vdash \Gamma^n, \Pi \Rightarrow \Delta^n, \Sigma$ and $d\mathcal{D} < d@_i\varphi$.*

Lemma .7 (Left reduction) *Let $\mathcal{D}_1 \vdash \Gamma \Rightarrow \Delta, @_i\varphi^n$ and $\mathcal{D}_2 \vdash @_i\varphi, \Pi \Rightarrow \Sigma$ and $d\mathcal{D}_1, d\mathcal{D}_2 < d@_i\varphi$, then we can construct a proof $\mathcal{D}$ such that $\mathcal{D} \vdash \Gamma, \Pi^n \Rightarrow \Delta, \Sigma^n$ and $d\mathcal{D} < d@_i\varphi$.*

They hold for SC with substitutive and reductive rules. Lemma 6 makes a reduction on the right, and lemma 7 on the left premiss of cut by induction on the height of respective proofs. The latter in the case of principal cut-formula applies lemma 6. Eventually, lemma 7 yields, by induction on proof-degree:

Theorem .8 *Every proof may be transformed into cut-free proof.* □

References

[1] Beeson, M., "Foundations of Constructive Mathematics," Springer, 1985.

[2] Blackburn, P., *Internalizing labelled deduction*, Journal of Logic and Computation **10/1** (2000), pp. 137–168.

[3] Blackburn, P. and M. Marx, *Tableaux for quantified hybrid logic*, in: *Tableaux 2002, LNAI 2381*, Springer, 2002 pp. 38–52.

[4] Blackburn, P. and B. ten Cate, *Pure extensions, proof rules and hybrid axiomatics*, Studia Logica **84/2** (2006), pp. 277–322.

[5] Bolander, T. and T. Braüner, *Tableau-based decision procedures for hybrid logic*, Journal of Logic and Computation **16/6** (2006), pp. 737–763.

[6] Braüner, T., "Hybrid Logic and its Proof-Theory," Springer, 2011.

[7] Ciabattoni, A., *Automated generation of analytic calculi for logics with linearity*, in: *Proceedings of CSL'04, vol. 3210*, LNCS, 2004 pp. 503–517.

[8] Feferman, S., *Definedness*, Erkenntniss **43** (1995), pp. 295–320.

[9] Fitting, M., *Modal logic should say more than it does*, in: *Computational Logic, Essays in Honor of Alan Robinson*, Cambridge, 1991 pp. 113–135.

[10] Fitting, M. and R. L. Mendelsohn, "First-Order Modal Logic," Kluwer, Dordrecht, 1998.

[11] Garson, J. M., "Modal Logic for Philosophers," Cambridge Univ. Press, 2006.

[12] Gazzari, R., *The calculus of natural calculation*, Studia Logica **submitted** (2020).

[13] Goldblatt, R., "Quantifiers, Propositions and Identity," Cambridge Univ. Press, 2011.

[14] Indrzejczak, A., "Natural Deduction, Hybrid Systems and Modal Logics," Springer, 2010.

[15] Indrzejczak, A., *Simple cut elimination proof for hybrid logic*, Logic and Logical Philosophy **25/2** (2016), pp. 129–141.

[16] Indrzejczak, A., *Cut-free modal theory of definite descriptions*, in: *Advances in Modal Logic 12*, College Publications, 2018 pp. 387–406.

[17] Indrzejczak, A., *Rule-generation theorem and its applications*, Bulletin of the Section of Logic **47/4** (2018), pp. 265–281.

[18] Indrzejczak, A., *A novel approach to equality*, Synthese **to appear** (2020).

[19] Indrzejczak, A. and M. Zawidzki, *Decision procedures for some strong hybrid logics*, Logic and Logical Philosophy **22/4** (2013), pp. 389–409.

[20] Jaśkowski, S., *On the rules of suppositions in formal logic*, Studia Logica **1** (1934), pp. 5–32.

[21] Manzano, M., M. Martins and A. Huertas, *A semantics in equational hybrid propositional type theory*, Bulletin of the Section of Logic **43/3-4** (2014), pp. 121–154.

[22] Manzano, M., M. Martins and A. Huertas, *Completeness in equational hybrid propositional type theory*, Studia Logica **107/6** (2019), pp. 1159–1198.

[23] Metcalfe, G., N. Olivetti and D. Gabbay, "Proof Theory for Fuzzy Logics," Springer, 2008.

[24] Orlandelli, E., *Labelled calculi for quantified modal logics with definite descriptions*, Journal of Logic and Computation **submitted** (2020).

[25] Orlandelli, E. and G. Corsi, *Labelled calculi for quantified modal logics with non-rigid and non-denoting terms*, in: *Proceedings of ARQNL 2018*, CEUR-WS.org, 2018 pp. 64–78.

[26] Restall, G., *Generality and existence 1: Quantification and free logic*, The Review of Symbolic Logic **12/1** (2019), pp. 354–378.

[27] Seligman, J., *Internalization: The case of hybrid logics*, Journal of Logic and Computation **11/5** (2001), pp. 671–689.

[28] Textor, M., *Towards a neo-brentanian theory of existence*, Philosophers' Imprint **17/6** (2017), pp. 1–20.

[29] Thomason, R. and R. Stalnaker, *Modality and reference*, Nous **2** (1968), pp. 359–372.

[30] Zawidzki, M., "Deductive Systems and the Decidability Problem for Hybrid Logics," Univ. of Lodz Press, Jagielonnian Univ. Press, 2013.