Labelled Proofs for Quantified Modal Logic

Alberto Artosi[1], Paola Benassi[2], Guido Governatori[2] and Antonino Rotolo[2]

[1] Department of Philosophy, University of Bologna, via Zamboni 38, 40126, +39-51-258333

[2] CIRFID, University of Bologna via Galliera 3, 40126, +39-51-277204
email: {artosi,pbenassi,governat,rotolo}@cirfid.unibo.it

Abstract. In this paper we describe a modal proof system arising from the combination of a tableau-like classical system, which incorporates a restricted ("analytic") version of the cut rule, with a label formalism which allows for a specialised, logic-dependent unification algorithm. The system provides a uniform proof-theoretical treatment of first-order (normal) modal logics with and without the Barcan Formula and/or its converse.

1 Introduction

This paper generalizes the KEM proof method for normal modal propositional logics described in [AG94] (and further refined and expanded in [ACG94b,ACG94a,Gov95]) to normal systems of first-order modal logic with and without the Barcan Formula and/or its converse. The critical feature of the original (propositional) method, besides its being based on a combination of tableau and natural deduction inference rules which allows for a suitably restricted ("analytic") use of the cut rule, is that it generates models and checks them for putative contradictions using a label scheme to bookkeep "world" paths. Briefly and informally, we work with an alphabet of constant and variable "world" symbols. A "world" label is a world-symbol or a "structured" sequence of world-symbols we call a "world-path". Constant and variable world-symbols can be viewed as denoting worlds and sets of worlds respectively (in a Kripke model), while a world-path conveys information about access between the worlds in it. We attach labels to signed formulas (i.e., formulas prefixed with a "T" or "F") to yield *labelled signed formulas* (*LS-formulas*). A LS-formula TA, i (FA, i) means that A is true (false) at the (last) world (on the path) i. In the course of the proof, labels are manipulated in a way closely related to the semantics of modal operators and "matched" using a specialized (logic-dependent) unification algorithm. That two structured labels i and k are unifiable means that they virtually represent the same path, i.e. any world which you could get to by the path i could be reached by the path k and vice versa. LS-formulas whose labels are unifiable turn out to be true (false) at the same world(s) relative to the accessibility relation that holds in the appropriate class of models.

As we show in this paper, such a formalism is readily extended to first-order versions of the usual normal modal logics by further labelling symbols with the

J.J. Alferes, L. M. Pereira and E. Orlowska(eds)
Logics in Artificial Intelligence.
LNCS 1126, pp. 70–86, 1996.

individuals in their associated domain. At this end, we introduce two more sets of formal symbols which play the role of "renamings" of the individual terms of the language. This allows us to characterise each of the variants of the first-order modal logics $K, D, T, S4, B, S5$ by using the familiar quantifier rules of the tableau method in combination with corresponding appropriate versions of the modal rules.

2 An Outline of Quantified Modal Logic

In what follows we assume a Modal First-Order Language (without function symbols) L defined in the usual way. Let $\mathfrak{C} = \{c_1, c_2, \dots\}$ and $\mathfrak{V} = \{x_1, x_2, \dots\}$ be the sets of individual symbols (resp. constants and variables) and $\mathfrak{P} = \{P_1, P_2, \dots\}$ the set of predicates of L. A system L of Quantified Modal Logic (QML) is constituted by

1. classical and modal propositional axioms;
2. $\forall x(A \to B) \to (A \to \forall x B(x))$, x not free in A;
3. $\forall y(\forall x A(x) \to A(y))$

and possibly by either of (or both) the following formulas (Barcan Formula and its Converse):

$$\forall x \Box A(x) \to \Box \forall x A(x) \tag{BF}$$

$$\Box \forall x A(x) \to \forall x \Box A(x) \tag{CBF}$$

All the systems of QML we shall be concerned with include *modus ponens*, *necessitation*, and *universal generalization*. For constant domains we have also *universal instantiation*.

A *First-Order Kripke Model* $\mathcal{M}$ is a 5-tuple $\langle \mathcal{W}, R, \mathcal{D}, e, v \rangle$ where $\mathcal{W}$ is a (non empty) set of possible worlds, R is the accessibility relation on $\mathcal{W}$, $\mathcal{D}$ is a (non empty) set of individuals, e is a mapping $e : \mathcal{W} \to \wp(\mathcal{D})$ which assigns to each possible world a domain of individuals, and v is the usual valuation function such that, for any $c_n \in \mathfrak{C}$ and any $w_i, w_j \in \mathcal{W}$, $v(c_n, w_i) = v(c_n, w_j)$ and, for any n-ary predicate $P_m \in \mathfrak{P}$ and any $w_i \in \mathcal{W}$, $v(P_m, w_i) \subseteq (e(w_i))^n$. Furthermore, formulas are evaluated classically (see [Kri63]).

3 *KEM* Language and Label Formalism

As usual with refutation methods, a *KEM*-proof of a formula A consists of attempting to construct a countermodel for A by assuming that A is false in some arbitrary model $\mathcal{M}$. In proving formulas of L we shall use *labelled signed formulas* (*LS*-formulas), i.e. expressions of the form SA, i where $S \in \{T, F\}$, A is a formula of L and i is a label.

The set $\Im$ of labels arises from two (non empty) sets $\Phi_C = \{w_1, w_2, \dots\}$ and $\Phi_V = \{W_1, W_2, \dots\}$ respectively of *constants* and *variable world symbols* through the following definition:

$$\Im = \bigcup_{1 \leq i} \Im_i \text{ where } \Im_i \text{ is :}$$
$$\Im_1 = \Phi_C \cup \Phi_V;$$
$$\Im_2 = \Im_1 \times \Phi_C;$$
$$\Im_{n+1} = \Im_1 \times \Im_n.$$

That is, a world-label is either (i) an element of the set Φ_C, or (ii) an element of the set Φ_V, or (iii) a path term (k', k), where (iiia) $k' \in \Phi_C \cup \Phi_V$ and (iiib) $k \in \Phi_C$ or $k = (i', i)$, where (i', i) is a label. From now on we shall use $i, j, k, \dots$ to denote arbitrary labels.

For any label $i = (k', k)$ we shall call k' the *head* of i, k the *body* of i, and denote them by $h(i)$ and $b(i)$ respectively. Notice that these notions are recursive (they correspond to projection functions): if $b(i)$ denotes the body of i, then $b(b(i))$ will denote the body of $b(i)$, $b(b(b(i)))$ will denote the body of $b(b(i))$; and so on. We shall call each of $b(i)$, $b(b(i))$, etc., a *segment* of i. Let $s(i)$ denote any segment of i (obviously, by definition every segment $s(i)$ of a label i is a label); then $h(s(i))$ will denote the head of $s(i)$.

For any label i, we define the length of i, $l(i)$, as the number of world-symbols in i, i.e. $l(i) = n \Leftrightarrow i \in \Im_n$. $s^n(i)$ will denote the segment of i of length n, i.e. $s^n(i) = s(i)$ such that $l(s(i)) = n$. We shall use $h^n(i)$ and i^n indifferently as abbreviations for $h(s^n(i))$.

For any label $i, l(i) > n$, we define the *countersegment-n* of i, as follows:

$$c^n(i) = h(i) \times (\cdots \times (h^k(i) \times (\cdots \times (h^{n+1}(i), w_0)))) \; (n < k < l(i))$$

where w_0 is a "dummy" label, i.e. a label not appearing in i (the context in which such a notion is applied will tell us what w_0 stands for).

Example 1. If $i = (w_4, (W_3, (w_3, (W_2, w_1))))$ then $l(i) = 5$, $s^3(i) = (w_3, (W_2, w_1))$ and its countersegment-3 is $c^3(i) = (w_4, (W_3, w_0))$; intuitively, $c^n(i)$ is what remains of i after deleting $s^n(i)$.

We shall call a label i *restricted* if $h(i) \in \Phi_C$, otherwise *unrestricted*.

Let us extend L with the following sets of individual symbols: $\mathfrak{T} = \{t_1, t_2, \dots\}$ (the set of *tokens*), and $\mathfrak{M} = \{m_1, m_2, \dots\}$ (the set of *marks*), which will be used in proving formulas of L. We shall denote by d's arbitrary elements of $\mathfrak{T} \cup \mathfrak{M}$. We stipulate that if $i \in \Im_1$ and $t_1, \dots, t_n \in \mathfrak{T}$, then $i[\![t_1, \dots, t_n]\!] \in \Im_1$. Herein we shall use $i[\![t_1, \dots, t_n]\!]$ to denote $(h(i)[\![t_1, \dots, t_n]\!], b(i)[\![t'_1, \dots, t'_m]\!])$.

As an intuitive explanation, we may think of a label $i \in \Phi_C$ as denoting a world (a *given* one), and a label $i \in \Phi_V$ as denoting a set of worlds (*any* world) in some Kripke model. A label $i = (k', k)$ may be viewed as representing a path from k to a (set of) world(s) k' accessible from k (i.e., from the world(s) denoted by k). Tokens occurring in a label may be thought of as "known" elements of the domain of the world(s) in the path represented by the label(s) they are attached to.

Example 2. The label $(W_1, w_1[\![t_1]\!])$ represents a path which takes us to the set W_1 of worlds accessible from w_1, and t_1 denotes an element of the domain of w_1; $(w_2, (W_1[\![t_2, t_3]\!], w_1)))$ represents a path which takes us to a world w_2 accessible via any world accessible from w_1, (i.e. accessible from the sub-path $(W_1[\![t_2, t_3]\!], w_1))$ and t_2, t_3 stand for individuals "shared" by the worlds denoted by W_1.

4 Unifications

A characteristic feature of *KEM* proof method is the use of rules for *label*-unifications in order to determine whether two labels denote the same world(s) under the appropriate accessibility conditions. In dealing with first-order normal modal logics we need rules for *terms* unification in order to determine whether two terms (from the sets $\mathfrak{T}$ and $\mathfrak{M}$) denote the same individual under the appropriate domain conditions. In this section we provide both kinds of unifications.

4.1 Label Unifications

KEM label unification scheme involves two kinds of unifications, respectively "high" and "low" unifications. An high unification, σ^L, is meant to mirror a single constraint on R, whereas a low unification, σ_L, is used to simulate the full range of conditions governing the accessibility relation which characterizes L. High and low unifications are defined respectively as follows.

High unifications: First of all we define a substitution in the usual way as a mapping

$$\sigma : \Phi_V \longrightarrow \Im \, .$$

For two labels i, k and a substitution σ, if σ is a unifier of i and k then we shall say that i and k are σ-*unifiable*. We shall (somewhat unconventionally) use $(i, k)\sigma$ to denote both that i and k are σ-unifiable and the result of their unification. On this basis we define several specialised, logic-dependent notions of σ^L-unification. In particular, the notion of two labels i, k being σ^K-, σ^D-, σ^T-, σ^4-,σ^B-unifiable is defined as follows:

$$(i,k)\sigma^K = (i,k)\sigma \quad \begin{array}{l}\text{at least one of } i \text{ and } k \text{ is restricted, and}\\ \forall n \leq l(i), (s^n(i), s^n(k))\sigma^K\end{array}$$

$$(i,k)\sigma^D = (i,k)\sigma$$

$$(i,k)\sigma^T = \begin{cases} (s^{l(k)}(i),k)\sigma & l(i) > l(k), \text{ and} \\ & \forall n \geq l(k), (h^n(i), h(k))\sigma = (h(i), h(k))\sigma \\ (i, s^{l(i)}(k))\sigma & l(k) > l(i), \text{ and} \\ & \forall n \geq l(i), (h(i), h^n(k))\sigma = (h(i), h(k))\sigma \end{cases}$$

$$(i,k)\sigma^4 = \begin{cases} c^{l(i)}(k) & l(k) > l(i), h(i) \in \Phi_V \text{ and} \\ & w_0 = (i, s^{l(i)}(k))\sigma \\ c^{l(k)}(i) & l(i) > l(k), h(k) \in \Phi_V \text{ and} \\ & w_0 = (s^{l(k)}(i), k)\sigma \end{cases}$$

$$(i,k)\sigma^B = \begin{cases} (b(b(i)), k)\sigma & \text{if } h(i) \in \Phi_V \text{ and} \\ & (h(i), h(k))\sigma = (h(b(b(i))), h(k))\sigma \\ (i, b(b(k)))\sigma & \text{if } h(k) \in \Phi_V \text{ and} \\ & (h(i), h(k))\sigma = (h(i), h(b(b(k))))\sigma \end{cases}$$

The notions of σ^D- and σ^K-unification are related respectively to the accessibility conditions for D and K. Thus, for example, $(w_2, (W_1, w_1))$ and $(W_3, (W_2, w_1))$ are σ^D- but not σ^K-unifiable (the segments (W_1, w_1), (W_2, w_1) are in fact not σ^K-unifiable) since a world accessible from w_1 might not exist due to the lack of seriality. This means that the "denotations" of W_1 and W_2 might be empty, which obviously makes their unification impossible. For the notion of σ^T-unification, take for example $i = (w_3, (W_1, w_1))$ and $k = (w_3, (W_2, (w_2, w_1)))$. Here $(W_2, w_3)\sigma = (w_3, w_3)\sigma$, then i and k σ^T-unify to $(w_3, (w_2, w_1))$. This intuitively means that the world w_3, accessible from a sub-path $s(k) = (W_2, (w_2, w_1))$, after the deletion of W_2 from k, is accessible from any path i which turns out to denote the same world(s) as $s(k)$, the step from w_2 to W_2 being irrelevant because of the reflexivity of R. For the notion of σ^4-unification take, for example, $i = (W_3, (w_2, w_1))$ and $k = (w_5, (w_4, (w_3, (W_2, w_1))))$. Here $s^{l(i)}(k) = (w_3, (W_2, w_1))$. Then i and k labels σ^4-unify to $(w_5, (w_4, (w_3, (w_2, w_1))))$ since $(i, s^{l(i)}(k))\sigma = ((W_3, (w_2, w_1)), (w_3, (W_2, w_1)))\sigma$. This intuitively means that all the worlds accessible from a sub-path $s^{l(i)}(k)$ are accessible from any path i which leads to the same world(s) denoted by $s^{l(i)}(k)$. For the notion of σ^B-unification notice, for example, that $i = (W_1, (w_2, w_1))$ and $k = w_1$ σ^B-unify to w_1 since $(W_1, w_1)\sigma = (w_1, w_1)\sigma$. This intuitively means that $b(b(i))$ and k denote the same world, and such a world is one of the worlds accessible by simmetry from $b(i)$.

Low Unifications: We are now able to combine the above unifications in a single low unification for $L = K, D, T, S4, B, S5$.

$$(i,k)\sigma_L = \begin{cases} (c^n(i), c^m(k))\sigma^{L_1 \cdots L_n} \\ (i,k)\sigma^{L_1 \cdots L_n} \end{cases}$$

where $w_0 = (s^n(i), s^m(k))\sigma_L$ and

$$(i,k)\sigma^{L_1 \cdots L_n} = \begin{cases} (i,k)\sigma^{L_1} \\ \vdots \\ (i,k)\sigma^{L_n} \end{cases}$$

where $L_1 \cdots L_n$ stand for the axioms characterising L.

For $S5$ we provide the following specialised σ_{S5}-unification:

$$(i,k)\sigma_{S5} = (h(i), h(k))\sigma \ .$$

We shall say that i *extends* k iff there exists an $s(i)$ such that either (i) $s(i) = k$ or (ii) $(s(i), k)\sigma_L$; and that i *extends immediately* k iff i extends k and $s(i) = b(i)$. We now provide a useful property of labels and unifications.

Lemma 1. *If* $(i,k)\sigma_L = l$ *then* $(i,l)\sigma_L$ *and* $(l,k)\sigma_L$*.*

Proof. The proof will be by induction on the number of applications of $\sigma^{L_1 \cdots L_n}$ in a σ_L-unification. Let n be the number of such applications. If $n = 1$ then we have to prove the property for $\sigma^{L_1 \cdots L_n}$ [3], which means

$$(i,k)\sigma^{L_1,\ldots,L_n} = l \Rightarrow (i,l)\sigma^{L_1,\ldots,L_n}, (k,l)\sigma^{L_1,\ldots,L_n} \ . \tag{1}$$

We then provide the definition of σ^{DT4}

$$(i,k)\sigma^{DT4} = \begin{cases} (i,k)\sigma^D & l(i) = l(k) \\ (i,k)\sigma^T & l(i) < l(k), h(i) \in \Phi_C \\ (i,k)\sigma^4 & l(i) < l(k), h(i) \in \Phi_V \end{cases}$$

At this point we prove the property stated in (1) by induction on the length of labels.

If $\min\{l(i), l(k)\} = 1$ then we assume that $l(i) = 1$ (the proof for $l(k) = 1$ is similar). 1) $i \in \Phi_C$. If also $l(k) = 1$, we apply σ^D; in every case, by obvious considerations about σ, $l = (i,k)\sigma^D = i$, but $(i,i)\sigma^D$ and $(i,k)\sigma^D$. If $l(k) > 1$ and $(i,k)\sigma^T$, then $l = (i,k)\sigma^T = (i, s^1(k))\sigma^T = i$, hence $(i,i)\sigma^D$ and $(i,k)\sigma^T$. If $l(k) > 1$ and $(i,k)\sigma^B$, then $l = (i,k)\sigma^B = (i, s^1(k))\sigma = i$, therefore $(i,i)\sigma^D$ and $(i,k)\sigma^B$. 2) $i \in \Phi_V$ then by the definition of σ it unifies with any label, in particular $(i,k)\sigma^D = k = l$, whence $(i,k)\sigma^D$ and $(k,k)\sigma^D$.

Let us suppose now that $\min\{l(i), l(k)\} = n > 1$, and that the property holds up to n for $\sigma^{L_1 \cdots L_n}$. Thus we have the following cases.

$L_1 \cdots L_n = D$ and $L_1 \cdots L_n = K$. If $l(i) = l(k)$ then $(i,k)\sigma^D = l$; by the inductive hypothesis $(b(i), b(l))\sigma^D$, $(b(k), b(l))\sigma^D$, $(h(i), h(l))\sigma^D$ and $(h(k), h(l))\sigma^D$; therefore $(i,l)\sigma^D$ and $(k,l)\sigma^D$. The proof for K follows from the fact that l contains only constants, which implies that each single element of i and k is either a variable or the constant occurring in the corresponding place in l.

$L_1 \cdots L_n = DT$. If $l(i) < l(k)$ and $(i,k)\sigma^T = l$, by the inductive hypothesis $(b(i), b(l))\sigma^D$, $(s^{l(b(i))}(k), b(l))\sigma^D$. By the definition of σ^T, we know that $l^n = (h(i), h(k))\sigma = (h(i), h^{l(i)}(k))\sigma)$; therefore $(i,l)\sigma^D$ and $(k,l)\sigma^T$. The case $l(i) = l(k)$ is the same as the case for D above.

$L_1 \cdots L_n = DT4$. If $l(i) < l(k)$ and $h(i) \in \Phi_V$, then $(i,k)\sigma^4 = c^{l(i)}(k)$ where $w_0 = (i, s^{l(i)}(k))\sigma$. By the inductive hypothesis and the definition of σ we have

[3] Hereafter, in order to shorten proofs, when we have to consider labels of different lengths, we shall assume, unless specified, the first to be the shorter. Obviously proofs for the other cases carry out in the same way.

$(i, s^{l(i)}(l))\sigma$ and $(s^{l(i)}(k), s^{l(i)}(l))\sigma$ and therefore $(i,l)\sigma^4$ and $(k,l)\sigma^D$. The other clauses of σ^{DT4} are respectively the cases for T and D above.

$L_1 \cdots L_n = DTB$. If $l(i) < l(k)$ and $(i,k)\sigma^B = l$, by inductive hypothesis $(b(i), b(l))\sigma^D$, $(s^{l(b(i))}(k), b(l))\sigma^D$; by the definition of σ^B, we know that $l^n = (h(i), h(k))\sigma = (h(i), h(b(b(i))))\sigma)$; therefore $(i,l)\sigma^D$ and $(k,l)\sigma^B$. The other cases of the σ^{DTB} are respectively the cases for T and D above.

We have thus proved the inductive base for the lemma. We can now assume that the lemma holds up to the n-th application of $\sigma^{L_1 \cdots L_n}$. By the definition of σ_L, $(s^n(i), s^m(k))\sigma_L = w_0 = s^l(l)$ and $(c^n(i), c^m(k)j)\sigma^{L_1 \cdots L_n} = c^l(l)$, but, by the inductive hypothesis, $(s^n(i), s^l(l))\sigma_L$ and $(s^m(k), s^l(l))\sigma_L$. By the property we have just proved for $\sigma^{L_1 \cdots L_n}$ $(c^n(i), c^l(l))\sigma^{L_1 \cdots L_n}$ and $(c^m(k), c^l(l))\sigma^{L_1 \cdots L_n}$, which implies $(i,l)\sigma_L$ and $(k,l)\sigma_L$.

For $S5$ we have $(i,k)\sigma_{S5}$ iff $(h(i), h(k))\sigma$, whence, if i is restricted, then $(i,k)\sigma_{S5} = hi = l$ and thus $(i,l)\sigma_{S5}$, i.e. $(h(i), h(i))\sigma$, and similarly for k; otherwise $(i,k)\sigma_{S5} = h(k) = l$, therefore for the same reason as in the previous case $(k,l)\sigma_{S5}$ and $(i,l)\sigma_{S5}$.

4.2 Term Unifications

As said before, in proving formulas of L we use two kinds of symbols — tokens and marks — associated to world domains. Therefore we have to determine, via an appropriate unification, whether two such symbols denote the same individual(s) relative to a given world. In order to deal with constant, increasing, decreasing and varying domains we introduce a domain dependent ρ-unification between terms.

Given a set of labels $\mathcal{L}$ the ρ-unification is just the usual unification with the constraint that an indexed mark $(m_n)_i$, $i \in \Im$, ρ-unifies with a term d iff either i) d is a token t_m attached to a label $k \in \mathcal{L}$ ($k[\![t_m]\!]$), such that $(i,k)\sigma_L$, or ii) d is a mark. Two indexed marks, $(m_p)_i$ and $(m_q)_k$, ρ-unify iff $(i,k)\sigma_L$. For constant domains we only require that a mark ρ-unifies with a token iff the token is attached to a label in $\mathcal{L}$. For varying domain in a broader sense the ρ-unification is defined formally as follows:

$$(d, d')\rho = \begin{cases} d & \text{if } d = d' \\ (d)_{(i,k)\sigma_L} & \text{if } d = (m)_i, d' = (m')_k, \text{ and } (i,k)\sigma_L \\ t & \text{if } d' = (m)_k \text{ and } d = t \in D(k\sigma_L^{\mathcal{L}}) \\ t' & \text{if } d = (m)_i \text{ and } d' = t' \in D(i\sigma_L^{\mathcal{L}}) \end{cases}$$

where $D(i\sigma_L^{\mathcal{L}})$ ($D(k\sigma_L^{\mathcal{L}})$) is the set of tokens extracted from the head of i (k) and from the head of the labels in $\mathcal{L}$ unifying with i (k).

5 Inference Rules

In displaying the rules of KEM we shall use Smullyan-Fitting $\alpha, \beta, \gamma, \delta, \nu, \pi$ unifying notation [Fit83]. As usual X^C will denote the *conjugate* of X, i.e. the result of changing the sign of X to its opposite. Two LS-formulas $X(d), i$ and $X^C(d'), k$ such that $(i,k)\sigma_L$ and $(d,d')\rho$ will be called *$\sigma_L\rho$-complementary*. We shall write a β-formula also as $[\beta_1, \beta_2]$.

Propositional Rules

$$\frac{\alpha, k}{\alpha_n, k}[n = 1, 2] \qquad (\alpha\text{-rules})$$

$$\frac{\begin{array}{c}[\beta_1(d^1), \beta_2(d^2)], k\\ \beta_n^C(d), l\end{array}}{\beta_{3-n}(d^{3-n}), (k,l)\sigma_L}[(k,l)\sigma_L, (d^n, d)\rho \text{ and } n = 1, 2] \qquad (\beta\text{-rules})$$

Rewriting Rules

$$\frac{X(c_n), i}{X(t_n), i[\![t_n]\!]} \qquad (\text{constant rewriting rule})$$

For constant domains t_n is always attached to i, whereas for varying domain in a broader sense it is attached to i iff $X(c_n)$ is atomic, i.e. $X(c_n) = TP(c_n)$ for some predicate P.

$$\frac{X(x_n), i}{X(m_n)_i, i} \qquad (\text{variable rewriting rule})$$

Quantifier Rules

$$\frac{\gamma, i[\![t_1, \dots, t_n]\!]}{\gamma_0(m_n)_i, i[\![t_1, \dots, t_n]\!]}[m_n \text{ new}] \qquad (\gamma\text{-rules})$$

$$\frac{\delta, i[\![t_1, \dots, t_n]\!]}{\delta_0(t_m), i[\![t_1, \dots, t_n, t_m]\!]}[h(i) \in \Phi_C \text{ and } t_m \text{ new}] \qquad (\delta\text{-rules})$$

$$\frac{\delta, i[\![t_1, \dots, t_n]\!]}{\delta_0(t_m), i[\![t_1, \dots, t_n]\!]}[h(i) \in \Phi_V \text{ and } t_m \text{ new}] \qquad (\delta\text{-rules})$$

Modal Rules

$$\frac{\nu, i}{\nu_0, (i', i)}[i' \in \Phi_V \text{ and new}] \qquad (\nu\text{-rules})$$

$$\frac{\pi, i}{\pi_0, (i', i)}[i' \in \Phi_C \text{ and new}] \qquad (\pi\text{-rules})$$

Modal Rules for Increasing Domains

$$\frac{\nu, i[\![t_1, \dots, t_n]\!]}{\nu_0, (i'[\![t_1, \dots, t_n]\!], i[\![t_1, \dots, t_n]\!])}[i' \in \Phi_V \text{ and new}] \qquad (\nu\text{I-rules})$$

$$\frac{\pi, i[\![t_1, \dots, t_n]\!]}{\pi_0, (i'[\![t_1, \dots, t_n]\!], i[\![t_1, \dots, t_n]\!])}[i' \in \Phi_C \text{ and new}] \qquad (\pi\text{I-rules})$$

Domains Rules

$$\frac{\begin{array}{c} i[t_1, \dots, t_n] \\ k[t'_1, \dots, t'_m] \end{array}}{(i,k)\sigma_L[\![t_1, \dots, t_n, t'_1, \dots, t'_m]\!]} \qquad \text{(Domains rule)}$$

$$\frac{\begin{array}{c} i[\![t_1, \dots, t_n]\!] \\ k[\![t'_1, \dots, t'_m]\!] \end{array}}{k[\![t_1, \dots, t_n, t'_1, \dots, t'_m]\!]}[k \text{ extends } i] \qquad \text{(Increasing domains rule)}$$

$$\frac{\begin{array}{c} i[\![t_1, \dots, t_n]\!] \\ k[\![t'_1, \dots, t'_m]\!] \end{array}}{k[\![t_1, \dots, t_n, t'_1, \dots, t'_m]\!]}[i \text{ extends } k] \qquad \text{(Decreasing domains rule)}$$

$$\frac{\begin{array}{c} i[\![t_1, \dots, t_n]\!] \\ k[\![t'_1, \dots, t'_m]\!] \end{array}}{k[\![t_1, \dots, t_n, t'_1, \dots, t'_m]\!]}[i \text{ immediately extends } k]$$

(Increasing Symmetric domains rule)

Structural Rules

$$\frac{\begin{array}{c} X(d), i \\ X^C(d'), k \end{array}}{\times(i,k)\sigma_L}[(i,k)\sigma_L \text{ and } (d,d')\rho] \qquad \text{(PNC)}$$

$$\frac{}{X, i \qquad X^C, i}[i \text{ restricted}] \qquad \text{(PB)}$$

When we split with respect to $X(m_n)_i$, after the application of PB, $(m_n)_i$ should be instantiated, in both branches, to the same token; X and X^C do not contain variables.

Here the α rules are just the familiar linear branch-expansion rules of the tableau method. In place of the usual tableau branching β rules we have a set of linear 2-premise β rules which represent well-known natural inference principles (such as disjunctive syllogism and its dual, modus ponens, and modus tollens). For example, from $\beta, k = TP(m_1)_{w_1} \to Q(t_1), (W_1, w_1[\![t_1]\!])$, where $\beta_1(d^1) = FP(m_1)_{w_1}$, $\beta_2(d^2) = TQ(t_1)$, and $\beta_1^C(d), l = TP(t_2), (w_2, w_1[\![t_1]\!])$ we get $\beta_2(d^2), (k,l)\sigma_L = TQ(t_1), (w_2, w_1[\![t_1]\!])$ since $((W_1, w_1[\![t_1]\!]), (w_2, w_1[\![t_1]\!]))\sigma_L$ and $((m_1)_{w_1}, t_2)\rho$. According to the rewriting rules, whenever constants or variables occur in an inference step, a "name" (respectively a token or a mark) is given to each of them. Technically, a rewriting substitution is applied to pick up individuals in a given domain, and possibly to attach them to labels. The γ and δ rules are the usual quantifier rules of the tableau method modified in such a way as to attach the instantiation of the quantified variable to the current label (notice that in the δ rules tokens cannot be attached to unrestricted labels according to their intuitive interpretation). The ν and π rules are as usual for constant and varying domains, whereas for increasing domains they take care of monotonicity. The domains rules remind Gabbay's visa rules [Gab94] and allow us to "move" individuals through worlds according to the domains conditions. "New" in the proviso for the modal and quantifier rules means "new to the branch". PB (for Principle of Bivalence) is a 0-premise branching rule which plays the role of the cut rule of the sequent calculus (intuitive meaning: a formula A is either true or false in any *given* world, whence the requirement that i should be restricted). In the course of proof search we shall use an "analytic" version of KEM where every application of PB is restricted to immediate sub-formulas of β formulas already occurring in the branch (for further discussion see [DM94,AG94]). PNC (for Principle of Non-Contradiction) corresponds to the familiar branch-closure rule of the tableau method, saying that from the occurrence of a pair of $\sigma_L\rho$-complementary LS-formulas $X(d), i$ and $X^C(d'), k$ on a branch, we may infer the closure ("$\times$") of the branch. The $(i,k)\sigma_L$ in the "conclusion" of PNC means that the contradiction holds "in the same world" for the same individual. Labels are manipulated, according to these rules, in such a way that (1) in all the inferences via an α rule the label of the premise carries over unchanged to the conclusion; (2) in all inferences via a β rule the labels and terms of the premises must be respectively σ_L- and ρ-unifiable, so that the conclusion inherits their unification (this reflects the obvious fact that classical inferences are valid only within a given world and with respect to given individuals); (3) in all inferences via a ν and π rule the label of the premise is immediately extended to a new (restricted or unrestricted) label according to the domain conditions; and (4) for K, PB is applied only to already existing restricted labels.

6 Examples

In this section we provide some example proofs. The notions of a KEM-tree and of a KEM-proof are as in the propositional case.

The following formula is $S4$-provable for varying domains.

1. $F\Diamond\exists x\Box(((Px \wedge Rx) \vee \Box Qx) \rightarrow \Box\forall y\Diamond(Qy \rightarrow Py))$	w_1
2. $F\exists x\Box(((Px \wedge Rx) \vee \Box Qx) \rightarrow \Box\ Ay\Diamond(Qy \vee Py))$	(W_1, w_1)
3. $F\Box(((Pm_{(W_1,w_1)} \wedge Rm_{(W_1,w_1)}) \vee \Box Qm_{(W_1,w_1)}) \rightarrow \Box\forall y\Diamond(Qy \vee Py))$	(W_1, w_1)
4. $F((Pm_{(W_1,w_1)} \wedge Rm_{(W_1,w_1)}) \vee \Box Qm_{(W_1,w_1)}) \rightarrow \Box\forall y\Diamond(Qy \vee Py)$	$(w_2, (W_1, w_1))$
5. $T(Pm_{(W_1,w_1)} \wedge Rm_{(W_1,w_1)}) \vee \Box Qm_{(W_1,w_1)}$	$(w_2, (W_1, w_1))$
6. $F\Box\forall y\Diamond(Qy \vee Py)$	$(w_2, (W_1, w_1))$
7. $F\forall y\Diamond(Qy \vee Py)$	$(w_3, (w_2, (W_1, w_1)))$
8. $F\Diamond(Qt_1 \vee Pt_1)$	$(w_3[\![t_1]\!], (w_2, (W_1, w_1)))$
9. $FQt_1 \vee Pt_1$	$(W_2, (w_3[\![t_1]\!], (w_2, (W_1, w_1))))$
10. FQt_1	$(W_2, (w_3[\![t_1]\!], (w_2, (W_1, w_1))))$
11. FPt_1	$(W_2, (w_3[\![t_1]\!], (w_2, (W_1, w_1))))$

12. $T\Box Qm_{(W_1,w_1)}$	$(w_2, (W_1, w_1))$	13. $F\Box Qm_{(W_1,w_1)}$	$(w_2, (W_1, w_1))$
14. $TQm_{(W_1,w_1)}$	$(w_4, (w_2, (W_1, w_1)))$	16. $TPm_{(W_1,w_1)} \wedge Rm_{(W_1,w_1)}$	$(w_2, (W_1, w_1))$
15. $\times$		17. $TPm_{(W_1,w_1)}$	$(w_2, (W_1, w_1))$
		18. $TRm_{(W_1,w_1)}$	$(w_2, (W_1, w_1))$
		19. $\times$	

The steps from 1 to 11 are straightforward. At this point we have a β formula but not β_1^C nor β_2^C, thus we apply PB w.r.t. $\Box Qm_{(W_1,w_1)}$ and $(w_2, (W_1, w_1))$.

In the left branch, we obtain 14 which is a formula $\sigma_{S4}\rho$-complementary of 10, since their labels σ_{S4}-unify and $(t_1, m_{(W_1,w_1)})\rho$, because $t_1 \in D((W_1, w_1)\sigma^{\mathcal{L}}_{S4})$, (i.e. $((W_1, w_1), (w_3, (w_2, (W_1, w_1))))\sigma_{S4}$). In the right branch, we get 17 which is, similarly, $\sigma_{S4}\rho$-complementary of 11. Notice that $m_{(w-1,w_1)}$ is instantiated in both branches to the same token t_1.

The following are KEM-proofs of the Barcan Formula and of its Converse.

1. $F\forall x\Box A(x) \rightarrow \Box\forall x A(x)$	w_1
2. $T\forall x\Box A(x)$	w_1
3. $F\Box\forall x A(x)$	w_1
4. $T\Box A(m_1)_{w_1}$	w_1
5. $F\forall x A(x)$	(w_2, w_1)
6. $TA(m_1)_{w_1}$	(W_1, w_1)
7. $FA(t_1)$	$(w_2[\![t_1]\!], w_1)$
8. $\times$	

The steps from 1 to 7 are straightforward. For decreasing domains, we apply to 7 the decreasing domains rule thus obtaining $(w_2[\![t_1]\!], w_1[\![t_1]\!])$. At this point $((W_1, w_1), (w_2[\![t_1]\!], w_1[\![t_1]\!]))\sigma_L$ and $((m_1)_{w_1}, t_1)\rho$ because $t_1 \in D(w_1\sigma^{\mathcal{L}}_L)$, so the tree is closed. Notice that $((m_1)_{w_1}, t_1)\rho$ holds also for constant domains, thus proving the formula for the corresponding systems.

1. $F\Box\forall x A(x) \rightarrow \forall x\Box A(x)$	w_1
2. $T\Box\forall x A(x)$	w_1
3. $F\forall x\Box A(x)$	w_1
4. $T\forall x A(x)$	(W_1, w_1)
5. $TA(m_1)_{(W_1,w_1)}$	(W_1, w_1)
6. $F\Box A(t_1)$	$w_1[\![t_1]\!]$
7. $FA(t_1)$	$(w_2, w_1[\![t_1]\!])$
8. $\times$	

The steps from 1 to 7 are straightforward. For increasing domains we apply to 7 the increasing domains rule thus obtaining $(w_2[\![t_1]\!], w_1[\![t_1]\!])$. At this point $((W_1, w_1), (w_2[\![t_1]\!], w_1[\![t_1]\!]))\sigma_L$ and $((m_1)_{w_1}, t_1)\rho$ because $t_1 \in D(w_1\sigma^{\mathcal{L}}_L)$, so the tree is closed. Notice that $((m_1)_{w_1}, t_1)\rho$ holds also for constant domains, thus proving the formula for the corresponding systems. It is important to note (and easy to verify) that the order of the applications of the modal and quantifiers rules leading to the nodes 5 and 6 is irrelevant, since such rules are wholly permutable (for the problem of order dependence which arises from the non-permutability of the usual modal and quantifiers tableau rules see [Wal90]).

7 Soundness and Completeness

Let $\mathcal{M} = \langle \mathcal{W}, R, \mathcal{D}, e, v\rangle$ be an L-model where $\mathcal{W} = \Phi_C$; R is a binary relation on $\mathcal{W}$; $\mathcal{D} = \mathfrak{T}$; e and v are as before. In particular, for any $c_k \in \mathfrak{C}$, $t_k \in \mathfrak{T}$, $x_k \in \mathfrak{V}$, $m_k \in \mathfrak{M}$ and any $w_j \in \mathcal{W}$, $v(c_k, w_j) = v(t_k, w_j)$ and $v(x_k, w_j) = v(m_k, w_j)$.

Let g be a function from $\Im$ to $\wp(\mathcal{W})$ thus defined:

$$g(i) = \begin{cases} h(i) = \{h(i)\} & \text{if } h(i) \in \Phi_C \\ h(i) = \{w_i \in \mathcal{W} : g(b(i))Rw_i\} & \text{if } h(i) \in \Phi_V \\ i = \mathcal{W} & \text{if } i \in \Phi_V \end{cases}$$

Let a be a function from $\mathfrak{T} \cup \mathfrak{M}$ to $\wp(\mathfrak{T})$ thus defined:

$$a(d) = \begin{cases} \mathfrak{T}_i \in \wp(\mathfrak{T}) & \text{if } d = (m_n)_i \\ \{t_n\} & \text{if } d = t_n \in \mathfrak{T} \end{cases}$$

where $\mathfrak{T}_i = e(g(i))$ if i is restricted, and $\mathfrak{T}_i = \cap e(g(i))$ otherwise.

Let r be a function from $\Im$ to R thus defined:

$$r(i) = \begin{cases} \emptyset & \text{if } l(i) = 1 \\ g(i^1)Rg(i^2), \ldots, g(i^{n-1})Rg(h(i)) & \text{if } l(i) = n > 1 \end{cases}$$

Let f be a function from LS-formulas to v thus defined:

$$f(SA, i) =_{def} v(A, w_j) = S$$

for all $w_j \in g(i)$.

Lemma 2. *For any $i, k \in \Im$ if $(i, k)\sigma_L$ then $g(i) \cap g(k) \neq \emptyset$.*

Proof. The proof is by induction on the number of applications of $\sigma^{L_1,\ldots,L_n}$ in σ_L. We need first to prove the following:

Lemma 3. *For any $i, k \in \Im$ if $(i, k)\sigma^L$ then $g(i) \cap g(k) \neq \emptyset$.*

Proof. The proof is by induction on the length of labels. If $\min\{l(i), l(k)\} = 1$, then at least one of i and k is either a constant or a variable, so that five cases will be present. By the definition of unifications i, k are either: i) two constants, or ii) a variable and a constant, or iii) two variables, or iv) a variable and a label, or v) a constant and a label.[4]

Case i) Two constants unify if and only if they are the same constant, and so $i = k$; therefore from the definition of g, $g(i) = g(k)$ and so $g(i) \cap g(k) \neq \emptyset$.

Case ii) If i (resp. k) is a variable and k (resp. i) is a constant, then $g(i) = \mathcal{W}$ and $g(k) \in \wp(\mathcal{W})$ therefore also in this case $g(i) \cap g(k) \neq \emptyset$.

Case iii) and iv) These cases are identical to the previous ones because: 1) $\mathcal{W}$ is not empty, and 2) the variable is mapped to $\mathcal{W}$ and the label to some world(s) in it.

Case v) This case implies that $(i, k)\sigma^T$ or $(i, k)\sigma^B$. Let us assume, for the sake of economy, that $l(i) = 1$ and $l(k) = n > 1$. If $(i, k)\sigma^T$, then for each $h(s(k))$ such that $l(s(k)) > 1$ either $h(s(k)) \in \Phi_V$, or $h(s(k)) = i$; therefore $r(k) =$

[4] Cases ii), iii), and iv) are not found in KEM proofs, but they are useful both for dealing with cases in the inductive step and for case v).

$iRk^2, \ldots, k^{n-1}Rk^n$. If $k^2 \in \Phi_V$, then k^2 denotes the set of worlds accessible from i; if $k^2 \in \Phi_C$, then $i = k$, but, through reflexivity $i \subseteq k^2$, so we take i as a representative of the set denoted by k^2, which implies iRk^3. We repeat the same argument until we arrive at iRk^n: if $k^n \in \Phi_C$, then $i = k^n$ and so they denote the same world; if $k^n \in \Phi_V$, then it denotes the set of worlds accessible from i; but i belongs to such a set, therefore, in all cases $g(i) \cap g(k) \neq \emptyset$. If $(i,k)\sigma^B$, then $h(k) \in \Phi_V$, $(i, h(k))\sigma$ and $(i, b(b(k)))\sigma$; moreover $r(k) = k^1Rk^2, k^2Rk^3$, but $k^1 = i$, and, by symmetry k^2Rk^1, which implies $k^1 \cap k^3 \neq \emptyset$, therefore $g(i) \cap g(k) \neq \emptyset$.

For the inductive step we have $\min\{l(i), l(k)\} = n > 1$. Let us assume inductively that the lemma is valid up to n; if $l(i) = l(k)$ we shall write i and k as $(h(i), b(i))$ and $(h(k), b(k))$, respectively. If $(i,k)\sigma^D$, by the definition of σ^D we get $(b(i), b(k))\sigma^D$, for which the lemma holds; let w_j be one of the worlds shared by $b(i)$ and $b(k)$, whence $w_jRh(i)$ and $w_jRh(k)$. We have now only to analyse what kind of labels are $h(i)$ and $h(k)$, which falls under the cases i), ii), and iii). Cases i) and ii) are the same as the inductive base. We have thus to examine case iii). Both $h(i)$ and $h(k)$ denotes the set of worlds accessible from w_j, but such a set is not empty because of the seriality of R. If $(i,k)\sigma^K$ we repeat the argument for D apart from cases iii), iv), and v) of the base which are not allowed in σ^K.

If $l(i) \neq l(k)$, we shall assume that $l(i) < l(k)$ (the case $l(k) < l(i)$ is dealt with in the same way). If $(i,k)\sigma^T$ and $h(i) \in \Phi_C$ then $(i, s^{l(i)}(k))\sigma^D$, therefore, combining the proofs of the previous case and case v) of the inductive base we obtain the desired result. If $h(i) \in \Phi_V$, then for all $k^n, n \leq l(i)$, $(h(i), h(k))\sigma = (h(i), k^n)\sigma$ which means $g(i) \cap g(s^n(k)) \neq \emptyset$, and in particular $g(i) \cap g(s^{l(i)}(k)) \neq \emptyset$.

If $(i,k)\sigma^4$ then $h(i) \in \Phi_V$ and $(b(i), s^{l(i)-1}(k))\sigma^D$, for which the inductive hypothesis holds; let w_j be such a shared world. $h(i)$ denotes all the worlds accessible from w_j, but, due to transitivity, the world(s) denoted by $h(k)$ belong(s) to $h(i)$ and so $g(i) \cap g(k) \neq \emptyset$.

If $(i,k)\sigma^B$ and $l(i) \leq l(k)$ then $h(k) \in \Phi_V$ and $(i, b(b(k)))\sigma$, for which the inductive hypothesis hold; let w_j be such a shared world. By repeating the same argument as for case v) of the base for B we get $g(i) \cap g(k) \neq \emptyset$.

We now return to the proof of the main lemma. If σ_L consists of a single step of $\sigma^{L_1 \cdots L_n}$, then $(i,k)\sigma_L = (i,k)\sigma^{L_1 \cdots L_n}$; by Lemma 3 we obtain $g(i) \cap g(k) \neq \emptyset$.

Let us assume, inductively, that the lemma holds up to n. If σ_L consists of $n+1$ applications of $\sigma^{L_1 \cdots L_n}$-unifications, then $(i,k)\sigma_L = (c^i(i), c^k(k))\sigma^{L_1 \cdots L_n}$ where $(s^i(i), s^k(k))\sigma_L$, which contains n applications of $\sigma^{L_1 \cdots L_n}$, and so the lemma holds for it. We can now repeat the argument of Lemma 3 with respect to $(c^i(i), c^k(k))\sigma^{L_1 \cdots L_n}$, proving thus that $g(i) \cap g(k) \neq \emptyset$.

For σ_{S5} the proof turns out to be the proof for the cases i), ii) and iii) of the inductive base of Lemma 3.

Lemma 4. *For any d, d', if $(d, d')\rho$ then $a(d) \cap a(d') \neq \emptyset$.*

Proof. If $d, d' \in \mathfrak{M}$ we have to check whether the labels, say i, k, attached to them are the same label or they σ_L-unify. In both cases, by Lemma 2 and the fact that the domains of the worlds are not empty we obtain the desired result. If $d = t_n$ and $t_n \in D(k\sigma_L^{\mathcal{L}})$, then t_n belongs to the domain of k which is the set $\mathfrak{T}_k$; therefore also in this case $a(d) \cap a(d') \neq \emptyset$. If $d = d'$ then $a(d) = a(d')$ and so $a(d) \cap a(d') \neq \emptyset$ trivially.

Lemma 5. *For any $i, k \in \Im$ and d, d', if $f(SA(d), i)$, $(i,k)\sigma_L$ and $(d,d')\rho$ then $f(SA(d'), (i,k)\sigma_L)$.*

Proof. Let us suppose that the lemma does not hold, so that $v(A(d), w_j) = S$ and $v(A(d'), w_h) = S^C$, for all $w_j \in g(i)$ and $w_h \in g(k)$. However, according to Lemma 2 and Lemma 4 $g(i) \cap g(k) \neq \emptyset$ and $a(d) \cap a(d') \neq \emptyset$, which means that there is a world $w_m \in g((i,k)\sigma_L)$ and an individual $t_n \in a((d,d')\rho)$ such that $v(A(t_n), w_m) = S$ and $v(A(t_n), w_m) = S^C$, thus obtaining a contradiction.

Theorem 1. $\models_L A \iff \vdash_L A$.

Proof. For a proof see, for example, [HC68,Gab76].

Theorem 2. $\vdash_L A \Rightarrow \vdash_{KEM(L)} A$.

Proof. The characteristic axioms of L and *modus ponens* are provable in KEM (see section 6 for a proof of the Barcan Formula and of its Converse, [Gov96] for a proof of some characteristic axioms and of *necessitation*, and [DM94] for a proof that *modus ponens* is a derived rule in the propositional fragment of KEM). Here we prove that *universal generalisation* is a derived rule of KEM.

$$
\begin{array}{llll}
 & 1.\ TAx \quad w_1 & & \\
2.\ T\forall xAx & w_1 & 3.\ F\forall xAx & w_1 \\
4.\ TA(m_1)_{w_1} & w_1 & 5.\ FAt_1 & w_1[\![t_1]\!] \\
 & & 6.\ \times &
\end{array}
$$

Theorem 3. $\vdash_{KEM(L)} A \Rightarrow \models_L A$.

Proof. The α-*rules* and PB are obviously sound rules in $\mathcal{M}$. For the β-*rules* and PNC: by the hypothesis $(l,k)\sigma_L$ and $(d,d')\rho$, then, by Lemma 1, $(i,(i,k)\sigma_L)\sigma_L$ and $(k,(i,k)\sigma_L)\sigma_L$ hence, by Lemma 5, the formulas involved have the same value in $g(i)$, $g(k)$ and $g((i,k)\sigma_L)$; after that these rules become rules of KE, and thus they are sound rules in $\mathcal{M}$.

For *Domains Rule*. If $(i,k)\sigma_L$ then $g(i) \cap g(k) \neq \emptyset$. We have thus to consider three cases:

Case i) $h(i), h(k) \in \Phi_C$; then $g(i) = g(k)$ and so $(i,k)\sigma_L[\![t_1,\ldots,t_n,t'_1,\ldots,t'_m]\!]$.

Case ii) $h(i) \in \Phi_V$ and $h(k) \in \Phi_C$ (or vice versa); then $g(i) \cap g(k) = g(k)$. Each $w_i \in g(i)$ is such that $w_i[\![t_1,\ldots,t_n]\!]$, and $g(k)$ is $g(k)[\![t'_1,\ldots,t'_m]\!]$, and so $(i,k)\sigma_L[\![t_1,\ldots,t_n,t'_1,\ldots,t'_m]\!]$.

Case iii) $h(i), (h(k)) \in \Phi_V$; in this case $g(i) \cap g(k) = g((i,k)\sigma_L)$. Any worlds $w_i \in g(i)$ and $w_k \in g(k)$ are such that $w_i[\![t_1,\ldots,t_n]\!]$ and $w_k[\![t'_1,\ldots,t'_m]\!]$, so $(i,k)\sigma_L[\![t_1,\ldots,t_n,t'_1,\ldots,t'_m]\!]$.

For *Increasing Domains Rule*. We know that k extends i, so $g(i)R^n g(k)$ or $g((i, s(k))\sigma_L)R^n g(k)$ by Lemma 2; however both cases implies $e(g(i)) \subseteq e(g(k))$, and so $k[\![t_1, \ldots, t_n, t'_1, \ldots, t'_m]\!]$.

The proofs for the *Decreasing- Increasing Symmetric- Domains Rule* are similar to that for *Increasing Domain Rule*.

For *Constant Rewriting Rule*. By the semantic conditions, for any $w_i \in \mathcal{W}$ and any c_n, t_n, $v(c_n, w_i) = v(t_n, w_i) \in a(t_n)$. For constant domains, since $v(t_n, w_i) = t_n \in e(g(i))$, then $i[\![t_n]\!]$ and so $f(SA(c_n), i) = f(SA(t_n), i[\![t_n]\!])$. For varying domains, by the definition of the valuation function, $f(SA(c_n), i) = f(SA(t_n), i[\![t_n]\!])$ iff $SA(c_n) = TP(c_n)$ for some predicate P.

For *Variable Rewritting Rule*. By the semantic conditions, for any $w_j \in g(i)$, $v(x_n, w_j) = v((m_n)_{w_j}, w_j)$. Moreover, $e(w_j) = \mathfrak{T}_{w_j}$ and $v((m_n)_{w_j}, w_j) = a((m_n)_{w_j}, w_j) = \mathfrak{T}_{w_j}$, therefore, $f(SA(x_n), i) = f(SA(m_n)_i, i)$.

For *δ-rules*. We show the proof only for $\delta = T\exists xA$ (the other case follows by the usual interdefinability of quantifiers). Let us suppose that $i \in \Phi_C$, $f(\delta, g(i[\![t_1, \ldots, t_n]\!])) = S$ and $f(\delta_0(t_m), g(i[\![t_1, \ldots, t_n, t_m]\!])) = S^C$, thus $\delta_0(t_m) = FA(t_m)$ and so we have $T\neg A(t_m)$. Since t_m is new to the branch, then $T\forall x \neg A(x)$, and so $T\neg\exists x A(x)$, contrary to the hypothesis. The proof for i unrestricted is similar.

For *ν-rules*. Let us suppose $\nu = T\Box A$; for all $w_j \in g(i)$ and for all $w_m \in g((i', i))$, $v(\Box A, w_j) = T$; but $v(\Box A, w_j) = T$ iff $\forall w_m : w_j R w_m, v(A, w_m) = T$, and $(\forall w_m : w_j R w_m, v(A, w_m) = T) = f(\nu_0, i')$ with i' unrestricted. The proof for π-rules is similar. For the νI- and the πI-rules it is sufficient to combine the above proofs for ν-rules, π-rules and *Increasing Domains Rule*.

From Theorems 1, 2, and 3 we obtain:

Theorem 4. $\vdash_{KEM(L)} A \iff \models_L A$.

8 Final Remarks

In the last ten years several theorem proving systems for first-order modal logic have been proposed. All suffer of severe limitations. For example, resolution [AM86] and translation [AE92,**?**] based methods are bound to resort to *ad hoc* methods of preprocessing the input formulas. Furthermore, resolution methods fail to provide a simple and uniform treatment of the full range of modal logics (see e.g. [AM86]). Sequent/tableau inference techniques [Fit88,**?**,**?**] avoid (in part, at least) these limitations (indeed [Fit88] tableau system with "branch modification" rules works only for non symmetric "cumulative" domain logics). However, both resolution and sequent/tableau inference rules fail to solve the problem associated with the non-permutability of the quantifier and modal rules (this holds true for both [AM86] and [Fit93] "prefixed" tableaux). Wallen's [Wal90] matrix proof (an extension of Bibel's classical connection) method is devised to overcome all these shortcomings. Its major drawback is that it yields proofs in a familiar, "natural deduction" style (e.g. in the form of sequent or tableau proofs) only derivatively and it works only for a few standard modal logics (it does not cover the "symmetric" B logics. Gent's [Gen93] generalization of Wallen's matrix proof method works for a wider range of logics but, unlike Wallen's, Gent's method requires translation of the modal formulas into a logic of restricted quantification). Of the theorem proving systems just mentioned, Jackson and Reichgelt's [JR89]'s sequent based proof method is the most similar to ours, in that it allows the label of the formulas occurring in the proof and of the terms chosen as the instantiation of the quantified variables (labels are attached to individuals to indicate in which worlds they are introduced) to be matched using a unification algorithm plus some pieces of "external" reasoning concerning the appropriate accessibility restrictions.

The interest in the system just presented is that it provides a uniform treatment of QML without normal-forming or translation procedures. Furthermore it offers a simple solution to the permutation problem which arises at the level of the usual (tableau and resolution) quantifier and modal rules by making the search space wholly insensitive to their application order. But (unlike Wallen's matrix proof method) it implements *directly* familiar, natural inference patterns (it is, however, also well suited as a framework for representing proofs discovered by means of connection matrix proof-search methods). Finally, its label unification scheme (unlike Jackson and Reichgelt's) avoids skolemization and recursively embodies the conditions on the accessibility relation for the various modal logics, thus dispensing proof search from any piece of "external" reasoning.

References

ACG94a. Alberto Artosi, Paola Cattabriga, and Guido Governatori. An Automated Approach to Normative Reasoning. In Joost Breuker, editor, *Artificial Normative Reasoning*, pages 132–145, Amsterdam, 1994. ECAI'94.

ACG94b. Alberto Artosi, Paola Cattabriga, and Guido Governatori. KED: A Deontic Theorem Prover. In C. Biagioli, G. Sartor, and D. Tiscornia, editors, *Workshop on Legal Application of Logic Programming*, ICLP'94, Firenze, IDG, 1994: 60–76.

AE92. Y. Auffray and P. Enjalbert. Modal Theorem Proving: An Equational Viewpoint. *Journal of Logic and Computation*, 2, 1992: 247–259.

AG94. Alberto Artosi and Guido Governatori. Labelled Model Modal Logic. In *Workshop on Automated Model Building*, CADE 12, Nancy, 1994: 11–17.

AM86. M. Abadi and Z. Manna. Modal Theorem Proving. In J. H.Siekmann (ed.), *Proceedings of 8th International Conference on Automated Deduction*, LNCS, Vol. 230, Springer Verlag, 1986: 172–189.

DM94. Marcello D'Agostino and Marco Mondadori. The Taming of the Cut. *Journal of Logic and Computation*, 4, 1994: 285–319.

Fit83. Melvin Fitting. *Proof Methods for Modal and Intuitionistic Logics*. Reidel, Dordrecht, 1983.

Fit88. M. Fitting. First-Order Modal Tableaux. *Journal of Automated Reasoning*, 4, 1988: 191–213.

Fit93. Melvin Fitting. Basic Modal Logic. In C.J. Hogger, Dov M. Gabbay, and J.A. Robinson (eds.), *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 1. Oxford University Press, Oxford, 1993: 368–448.

Gab76. Dov M. Gabbay. *Investigations in Modal and Tense Logics*. Reidel, Dordrecht, 1976.

Gab94. Dov M. Gabbay. Classical vs Non-Classical Logics. In C.J. Hogger, Dov M. Gabbay, and J.A. Robinson (eds), *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 2, Oxford University Press, Oxford, 1994: 359–500.

Gen93. Ian Gent. Theory Matrices (for Modal Logics) Using Alphabetical Monotonicity. *Studia logica*, 52, 1993: 233–257.

Gov95. Guido Governatori. Labelled Tableaux for Multi-Modal Logics. In P. Baumgartner, R. Hähnle and J. Posegga (eds.), *Theorem Proving with Analytic Tableaux and Related Methods*, LNAI, Vol. 918, Springer-Verlag, Berlin, 1995: 79–94.

Gov96. Guido Governatori. A Duplication and Loop Checking Free System for S4. In P. Miglioli, U. Moscato, D. Mundici and M. Ornaghi (eds.) *$5^t h$ Workshop on Theoprem Proving with Analytic tableaux and Related Methods (Short Papers)*,Technical report 154–96, Università di Milano, 1996: 19–32.

HC68. G.E. Hughes and M.J. Cresswell. *An Introduction to Modal Logic*. Methuen, London, 1968.

JR89. Peter Jackson and Han Reichgelt. *Logic-Based Knowledge Representation*. Chapter *A General Proof Method for Modal Predicate Logic*, MIT Press, Cambridge Mass., 1989: 177-228.

Kri63. Saul Kripke. Semantical Considerations on Modal Logics. *Acta Philosophica Fennica*, 16, 1963: 83–94.

Ohl91. Hans Jürgen Ohlbach. Semantic Based Translation Methods for Modal Logics. *Journal of Logic and Computation*, 1, 1991: 691–746.

Wal90. L. Wallen. *Automated Deduction in Nonclassical Logics*. MIT Press, Cambridge Mass., 1990.