# Multimodal Logic Programming and Its Applications to Modal Deductive Databases

Linh Anh Nguyen

Institute of Informatics, University of Warsaw
ul. Banacha 2, 02-097 Warsaw, Poland
nguyen@mimuw.edu.pl

September 29, 2003

**Abstract**

We give a general framework for developing the least model semantics, fixpoint semantics, and SLD-resolution calculi for logic programs in multimodal logics whose frame restrictions consist of the conditions of seriality (i.e. $\forall x\ \exists y\ R_i(x, y)$) and some classical first-order Horn formulas. Our approach is direct and no restriction on occurrences of $\Box_i$ and $\Diamond_i$ is required. We apply the framework for a large class of basic serial multimodal logics, which are parameterized by an arbitrary combination of generalized versions of axioms $T$, $B$, 4, 5 (in the form, e.g., $4 : \Box_i\phi \to \Box_j\Box_k\phi$) and $I : \Box_i\phi \to \Box_j\phi$. Another part of the work is devoted to programming in multimodal logics intended for reasoning about multi-degree belief, for use in distributed systems of belief, or for reasoning about epistemic states of agents in multi-agent systems. For that we also use the framework, and although these latter logics belong to the mentioned class of basic serial multimodal logics, the special SLD-resolution calculi proposed for them are more effective. We also apply the obtained results for multimodal deductive databases. We present modal relational algebras and advanced computational methods like the magic-set transformation for multimodal deductive databases.

## 1 Introduction

Modal and temporal logics are used to reason about knowledge, belief, actions, changes, etc. Many authors have proposed modal and temporal extensions for logic programming (see [31, 16] for surveys[1]). In modal logic programming, Balbiani et al [4] have given a declarative semantics and an SLD-resolution calculus for a class of modal logic programs in the logics $KD$, $T$, and $S4$. The work requires that the modal operator $\Box$ does not occur in bodies of programs and goals. In [7], Baldoni et al have given a framework for developing declarative and operational semantics for logic programs in multimodal logics which have axioms of the form $[t_1]\ldots[t_n]\phi \to [s_1]\ldots[s_m]\phi$, where $[t_i]$ and $[s_j]$ are universal modal operators parameterized by terms $t_i$ and $s_j$, respectively. In that work, existential modal operators are disallowed in programs and goals. The mentioned works directly use modal formulas. An alternative approach is based on translation of modal logic programs to classical logic programs. Debart et al [14] have applied a functional translation technique for logic programs in multimodal logics which have a finite number of modal operators $\Box_i$ and $\Diamond_i$ of any type among $KD$, $KT$, $KD4$, $KT4$, $KF$ and interaction axioms of the form $\Box_i\phi \to \Box_j\phi$. The technique is similar to the one used in Ohlbach's resolution calculus for modal logics [30]. Extra parameters are added to predicate symbols to represent paths in the Kripke model, and special unification algorithms are used to deal with them. In [29], Nonnengart has proposed an approach based on relational and functional translation. The approach uses accessibility relations for translated programs, but with optimized clauses for representing properties of the accessibility relations, and it does not modify unification. Nonnengart [29] has applied the approach for modal logic programs in all of the basic

[1]The works [29, 7, 28] on modal logic programming are not covered by the surveys.

serial monomodal logics $KD$, $T$, $KDB$, $B$, $KD4$, $S4$, $KD5$, $KD45$, and $S5$. He has also given an example in a multimodal logic of type $KD45$.

In our recent work [28], we have developed the least model semantics, a fixpoint semantics, and an SLD-resolution calculus in a direct way for modal logic programs in all of the above-mentioned serial monomodal logics. We have also extended the SLD-resolution calculus for the almost serial monomodal logics $KB$, $K5$, $K45$, and $KB5$. Important properties of our approach, in comparison with [4, 7], are that no restriction on occurrences of $\Box$ and $\Diamond$ is required and the semantics are formulated closely to the style of classical logic programming (as in Lloyd's book [22]). One of the main goals of this work is to extend the results and generalize the methods of our mentioned work for multimodal logics. Note that, in *multimodal* logic programming, the work [7] by Baldoni et al considers only so called inclusion multimodal logics and programs without existential modal operators; and using the translation approach, the work [14] by Debart et al does not consider symmetric multimodal logics (i.e. the ones with the axiom $B$ or 5).

In this work, we give a general framework for developing the least model semantics, fixpoint semantics, and SLD-resolution calculi for logic programs in multimodal logics whose frame restrictions consist of the conditions of seriality (i.e. $\forall x\ \exists y\ R_i(x,y)$) and some classical first-order Horn formulas. Our approach is direct and no restriction on occurrences of $\Box_i$ and $\Diamond_i$ is required. We apply the framework for a large class of basic serial multimodal logics, which are parameterized by an arbitrary combination of generalized versions of axioms $T$, $B$, 4, 5 (in the form, e.g., $4 : \Box_i\phi \to \Box_j\Box_k\phi$) and $I : \Box_i\phi \to \Box_j\phi$. Another part of the work is devoted to programming in multimodal logics intended for reasoning about multi-degree belief, for use in distributed systems of belief, or for reasoning about epistemic states of agents in multi-agent systems. For that we also use the framework, and although these latter logics belong to the mentioned class of basic serial multimodal logics, the special SLD-resolution calculi proposed for them are more effective. We also apply the obtained results for multimodal deductive databases. We present modal relational algebras and advanced computational methods like the magic-set transformation for multimodal deductive databases.

To illustrate our approach of defining semantics for multimodal logic programs, let us consider an example. Let the base logic be the multimodal logic $KD_{(m)}$, $G = \leftarrow \Diamond_1 t(x)$ be a goal and $P$ the program consisting of the following clauses:

$$\begin{aligned}
\phi_1 &= \Diamond_1 p(a) \leftarrow \\
\phi_2 &= \Box_1(\,\Box_2 q(x) \leftarrow p(x)\,) \\
\phi_3 &= \Box_1(\,\Diamond_2 r(x) \leftarrow p(x), \Box_2 q(x)\,) \\
\phi_4 &= \Box_1\Box_2(\,s(x) \leftarrow q(x), r(x)\,) \\
\phi_5 &= \Box_1(\,t(x) \leftarrow \Diamond_2 s(x)\,)
\end{aligned}$$

When building a $KD_{(m)}$-model graph $M$ for $P$, to realize $\phi_1$ at the actual world $\tau$ we connect $\tau$ to a world $w$ via the accessibility relation $R_1$ and add $p(a)$ to $w$. The edge connecting $\tau$ to $w$ is created due to $\Diamond_1 p(a)$, so we can label it by $\langle p(a)\rangle_1$ (a labeled form of $\Diamond_1$). The world $w$ can be identified by $\tau$ and the edge from $\tau$, so it can be denoted by the sequence $\tau\langle p(a)\rangle_1$. If we denote $\tau$ by the empty sequence then $w = \langle p(a)\rangle_1$. Apart from building $M$, we want to represent the model corresponding to $M$ by a set $I$ of *atoms*. To keep the information that $p(a)$ is true at $w$, we add the atom $\langle p(a)\rangle_1 p(a)$ to $I$. To realize $\phi_2$ at $\tau$, $\Box_2 q(x) \leftarrow p(x)$ is added to $w$, and then $\Box_2 q(a)$ is also added to $w$. To keep the fact that $\Box_2 q(a)$ belongs to $w$, we add $\langle p(a)\rangle_1\Box_2 q(a)$ to $I$. Note that $I$ contains both $\langle p(a)\rangle_1 p(a)$ and $\langle p(a)\rangle_1\Box_2 q(a)$. Apply the rule $\phi_3$ to $I$, then $I$ should contain also $\langle p(a)\rangle_1\Diamond_2 r(a)$, which is then replaced by $\langle p(a)\rangle_1\langle r(a)\rangle_2 r(a)$ due to a similar reason as for $\phi_1$. Since $I$ contains both $\langle p(a)\rangle_1\Box_2 q(a)$ and $\langle p(a)\rangle_1\langle r(a)\rangle_2 r(a)$, after applying $\phi_4$, $I$ should contain also $\langle p(a)\rangle_1\langle r(a)\rangle_2 s(a)$. Finally, applying $\phi_5$ to $I$, we get also $\langle p(a)\rangle_1 t(a)$. In general, instead of building a model graph for $P$ we can build such a set $I$ of atoms, which is called a *model generator*. The set $I_{KD_{(m)},P} = \{\langle p(a)\rangle_1 p(a), \langle p(a)\rangle_1\Box_2 q(a), \langle p(a)\rangle_1\langle r(a)\rangle_2 r(a), \langle p(a)\rangle_1\langle r(a)\rangle_2 s(a), \langle p(a)\rangle_1 t(a)\}$ is the least set of ground atoms which can be derived from $P$ in $KD_{(m)}$ in this way. This set is obtained as the least fixpoint of a certain operator $T_{KD_{(m)},P}$ and is called the *least* $KD_{(m)}$*-model generator* of $P$.

Given a model generator $I$, we can construct the *standard* $KD_{(m)}$*-model* for it by building a model graph. During the construction, to realize a formula $\langle E\rangle_i\phi$ at a world $w$, where $E$ is a ground classical atom, we connect $w$ via the accessibility relation $R_i$ to the world identified by the sequence $w\langle E\rangle_i$ and add $\phi$ to that world. We realize a formula $\Box_i\phi$ at a world $w$ by adding $\phi$ to every world reachable from $w$ via $R_i$. To guarantee the constructed model graph to be the smallest, each new world is connected via each accessibility relation to an empty world at the time of its creation. It can be shown that the standard $KD_{(m)}$-model of $I_{KD_{(m)},P}$ is a least $KD_{(m)}$-model of $P$.

Now let us give an SLD-refutation of $P \cup \{G\}$ in $KD_{(m)}$. By the content of $I_{KD_{(m)},P}$, the computed answer should be $\{x/a\}$. The SLD-refutation should trace back the process of deriving the atom $\langle p(a)\rangle_1 t(a)$ of $I_{KD_{(m)},P}$ from $P$. As a $KD_{(m)}$-resolvent of $G$ and $\phi_5$, we derive a new goal $G_1 = \leftarrow \Diamond_1\Diamond_2 s(x)$. As a $KD_{(m)}$-resolvent of $G_1$ and $\phi_4$, we derive the goal $G_2 = \leftarrow \Diamond_1\Diamond_2(q(x) \wedge r(x))$. This goal is not desired, as it contains a formula but not atoms in its body. To overcome this problem, the (existential) modality $\Diamond_1\Diamond_2$ should be fixed first, e.g., to become $\langle X\rangle_1\langle Y\rangle_2$, then the goal $G_2$ can be rewritten to $\leftarrow \langle X\rangle_1\langle Y\rangle_2 q(x), \langle X\rangle_1\langle Y\rangle_2 r(x)$. The labeling should be done in two steps as follows: the goal $G = \leftarrow \Diamond_1 t(x)$ is first replaced by $G' = \leftarrow \langle X\rangle_1 t(x)$, the next goal in the derivation is $G_1 = \leftarrow \langle X\rangle_1\Diamond_2 s(x)$, which is then replaced by $G'_1 = \leftarrow \langle X\rangle_1\langle Y\rangle_2 s(x)$, and then $G_2 = \leftarrow \langle X\rangle_1\langle Y\rangle_2 q(x), \langle X\rangle_1\langle Y\rangle_2 r(x)$ is derived from $G'_1$ and $\phi_4$. We can then strengthen $G_2$ to $G_3 = \leftarrow \langle X\rangle_1\Box_2 q(x), \langle X\rangle_1\langle Y\rangle_2 r(x)$. Resolving $G_3$ with $\phi_2$, we obtain $G_4 = \leftarrow \langle X\rangle_1 p(x), \langle X\rangle_1\langle Y\rangle_2 r(x)$. Now resolve $G_4$ with $\phi_1$. As explained in the construction of $I_{KD_{(m)},P}$, the atom $\Diamond_1 p(a)$ in the head of $\phi_1$ can be treated as $\langle p(a)\rangle_1 p(a)$. Thus, resolving $G_4$ with $\phi_1$ results in $G_5 = \leftarrow \langle p(a)\rangle_1\langle Y\rangle_2 r(a)$ and an mgu $\{x/a, X/p(a)\}$. Further steps are given in the following table. Note that for simplicity we have ignored renaming variables.

| Goals | Input Clauses | Mgu′s |
|---|---|---|
| $G\ = \leftarrow \Diamond_1 t(x)$ | | |
| $G' = \leftarrow \langle X\rangle_1 t(x)$ | | |
| $G_1 = \leftarrow \langle X\rangle_1\Diamond_2 s(x)$ | $\phi_5$ | $\varepsilon$ |
| $G'_1 = \leftarrow \langle X\rangle_1\langle Y\rangle_2 s(x)$ | | |
| $G_2 = \leftarrow \langle X\rangle_1\langle Y\rangle_2 q(x), \langle X\rangle_1\langle Y\rangle_2 r(x)$ | $\phi_4$ | $\varepsilon$ |
| $G_3 = \leftarrow \langle X\rangle_1\Box_2 q(x), \langle X\rangle_1\langle Y\rangle_2 r(x)$ | | |
| $G_4 = \leftarrow \langle X\rangle_1 p(x), \langle X\rangle_1\langle Y\rangle_2 r(x)$ | $\phi_2$ | $\varepsilon$ |
| $G_5 = \leftarrow \langle p(a)\rangle_1\langle Y\rangle_2 r(a)$ | $\phi_1$ | $\{x/a, X/p(a)\}$ |
| $G_6 = \leftarrow \langle p(a)\rangle_1 p(a), \langle p(a)\rangle_1\Box_2 q(a)$ | $\phi_3$ | $\{x_6/a, Y/r(a)\}$ |
| $G_7 = \leftarrow \langle p(a)\rangle_1\Box_2 q(a)$ | $\phi_1$ | $\varepsilon$ |
| $G_8 = \leftarrow \langle p(a)\rangle_1 p(a)$ | $\phi_2$ | $\{x_8/a\}$ |
| empty clause | $\phi_1$ | $\varepsilon$ |

This work is organized as follows: In Section 2, we give basic definitions for multimodal logics and introduce labeled modal operators together with their semantics. In Section 3, we define the MProlog language for multimodal logic programming and show that it is as expressive as the general modal Horn fragment. In Section 4, we give a framework for multimodal logic programming: we define fixpoint semantics, the least model semantics, and SLD-resolution for MProlog programs in a general level, giving a list of input definitions and a list of expected lemmas and theorems. Soundness and completeness of SLD-resolution are proved using lemmas that can be easily checked for concrete schemata of semantics of MProlog. The main theorem remained to be proved is the one stating that the fixpoint semantics coincides with the least model semantics. In Section 5, we use the framework to specify a schema for semantics of MProlog in the mentioned class of basic serial multimodal logics and prove its correctness. With the same style, in Section 6 (resp. 7) we give schemata for semantics of MProlog in multimodal logics intended for reasoning about multi-degree belief (resp. for use in multi-agent systems). In Section 8, we study the case when existential modal operators are disallowed in MProlog programs and goals. Section 9 is devoted to multimodal deductive databases: we define the query language MDatalog, prove its PTIME data complexity in some logics, and present modal relational algebras and computational methods. Discussion and conclusion are given in Section 10.

# 2 Preliminaries

## 2.1 Definitions for Quantified Multimodal Logics

A language for quantified multimodal logics is an extension of the language of the classical predicate logic with the modal operators $\Box_i$ and $\Diamond_i$, for $1 \leq i \leq m$ (where $m$ is fixed). If $m = 1$ then we ignore subscript $i$ beside $\Box$ and $\Diamond$. The modal operators $\Box_i$ and $\Diamond_i$ can take various meanings. For example, $\Box_i$ can stand for "it is believed with degree $i$" and $\Diamond_i$ for "it is possible at degree $i$". In this case we have logics for reasoning about multi-degree belief. As another example, $\Box_i$ stands for "the agent $i$ believes" and $\Diamond_i$ for "it is considered possible by agent $i$", and we have logics for multi-agent systems. Operators $\Box_i$ are called universal modal operators, while $\Diamond_i$ existential modal operators.

**Definition 2.1** A *term* is defined inductively as follows: a variable is a term; a constant symbol is a term; if $f$ is an $n$-ary function symbol and $t_1, \ldots, t_n$ are terms, then $f(t_1, \ldots, t_n)$ is a term.

**Definition 2.2** A (well-formed modal) *formula* is defined inductively as follows:

- If $p$ is an $n$-ary predicate symbol and $t_1, \ldots, t_n$ are terms, then $p(t_1, \ldots, t_n)$ is a formula, called a *classical atom*.
- If $\phi$ and $\psi$ are formulas, then so are $(\neg\phi)$, $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \to \psi)$, $(\Box_i\phi)$, and $(\Diamond_i\psi)$.
- If $\phi$ is a formula and $x$ is a variable, then $(\forall x.\phi)$ and $(\exists x.\phi)$ are formulas.

We also write $\phi \equiv \psi$ for $(\phi \to \psi) \wedge (\psi \to \phi)$.

The terminologies "free occurrence of variable", "closed formula", "ground term", "ground formula" are defined as usual as for the classical predicate logic.

The set of all ground terms is called the *Herbrand universe* and denoted by $\mathcal{U}$. The set of all ground classical atoms is called the *Herbrand base* and denoted by $\mathcal{B}$.

If $\phi$ is a formula, then by $\forall(\phi)$ we denote the *universal closure* of $\phi$, which is the closed formula obtained by adding a universal quantifier for every variable having a free occurrence in $\phi$. Similarly, $\exists(\phi)$ denotes the *existential closure* of $\phi$, which is obtained by adding an existential quantifier for every variable having a free occurrence in $\phi$.

The *modal depth* of a formula $\phi$, denoted by $mdepth(\phi)$, is the maximal nesting depth of modalities occurring in $\phi$. For example, the modal depth of $\Box_i(\Diamond_j p(x) \vee \Box_k q(y))$ is 2.

We now define Kripke models, model graphs, and the satisfaction relation.

**Definition 2.3** A *Kripke frame* is a tuple $\langle W, \tau, R_1, \ldots, R_m\rangle$, where $W$ is a nonempty set of possible worlds, $\tau \in W$ is the *actual world*, and $R_i$ is a binary relation on $W$, called the *accessibility relation* for the modal operators $\Box_i$, $\Diamond_i$. If $R_i(w, u)$ holds then we say that the world $u$ is accessible from the world $w$ via $R_i$.

A frame $\langle W, \tau, R_1, \ldots, R_m\rangle$ is said to be *connected* if every one of its worlds is directly or indirectly accessible from the actual world via the accessibility relations, i.e. for every $w \in W$ there exist $w_0 = \tau, w_1, \ldots, w_{k-1}, w_k = w$ with $k \geq 0$ such that $(w_i, w_{i+1}) \in R_1 \cup \ldots \cup R_m$ for all $0 \leq i < k$.

**Definition 2.4** A *fixed-domain Kripke model with rigid terms*, hereafter simply called a Kripke model or just a model, is a tuple $M = \langle D, W, \tau, R_1, \ldots, R_m, \pi\rangle$, where $D$ is a set called the *domain*, $\langle W, \tau, R_1, \ldots, R_m\rangle$ is a Kripke frame, and $\pi$ is an interpretation of constant symbols, function symbols and predicate symbols. For a constant symbol $a$, $\pi(a)$ is an element of $D$, denoted also by $a^M$. For an $n$-ary function symbol $f$, $\pi(f)$ is a function from $D^n$ to $D$, denoted also by $f^M$. For an $n$-ary predicate symbol $p$ and a world $w \in W$, $\pi(w)(p)$ is an $n$-ary relation on $D$, denoted also by $p^{M,w}$.

**Definition 2.5** A *model graph* is a tuple $\langle W, \tau, R_1, \ldots, R_m, H\rangle$, where $\langle W, \tau, R_1, \ldots, R_m\rangle$ is a Kripke frame and $H$ is a function that maps each world of $W$ to a set of formulas.

Every model graph $\langle W, \tau, R_1, \ldots, R_m, H\rangle$ corresponds to a Herbrand model $M = \langle \mathcal{U}, W, \tau, R_1, \ldots, R_m, \pi\rangle$ specified by: $c^M = c$, $f^M(t_1, \ldots, t_n) = f(t_1, \ldots, t_n)$, and $p^{M,w}(t_1, \ldots, t_n) \equiv (p(t_1, \ldots, t_n) \in H(w))$, where $t_1, \ldots, t_n$ are ground terms. We will sometimes treat a model graph as its corresponding model.

**Definition 2.6** Let $M$ be a Kripke model. A *variable assignment* (w.r.t. $M$) is a function that maps each variable to an element of the domain of $M$. The value of a term $t$ w.r.t. a variable assignment $V$ is denoted by $V(t)$ and defined as follows: If $t$ is a constant symbol $a$ then $V(t) = a^M$; if $t$ is a variable $x$ then $V(t) = V(x)$; if $t$ is $f(t_1, \ldots, t_n)$ then $V(t) = f^M(V(t_1), \ldots, V(t_n))$.

**Definition 2.7** Given some Kripke model $M = \langle D, W, \tau, R_1, \ldots, R_m, \pi\rangle$, some variable assignment $V$, and some world $w \in W$, the *satisfaction relation* $M, V, w \vDash \zeta$ for a formula $\zeta$ is defined as follows:

$$
\begin{array}{lll}
M, V, w \vDash p(t_1, \ldots, t_n) & \text{iff} & p^{M,w}(V(t_1), \ldots, V(t_n)); \\
M, V, w \vDash \neg\phi & \text{iff} & M, V, w \nvDash \phi; \\
M, V, w \vDash \phi \wedge \psi & \text{iff} & M, V, w \vDash \phi \text{ and } M, V, w \vDash \psi; \\
M, V, w \vDash \phi \vee \psi & \text{iff} & M, V, w \vDash \phi \text{ or } M, V, w \vDash \psi; \\
M, V, w \vDash \phi \to \psi & \text{iff} & M, V, w \nvDash \phi \text{ or } M, V, w \vDash \psi; \\
M, V, w \vDash \Box_i\phi & \text{iff} & \text{for all } v \in W \text{ such that } R_i(w, v),\ M, V, v \vDash \phi; \\
M, V, w \vDash \Diamond_i\phi & \text{iff} & \text{for some } v \in W,\ R_i(w, v) \text{ and } M, V, v \vDash \phi; \\
M, V, w \vDash \forall x.\phi & \text{iff} & \text{for all } a \in D,\ (M, V', w \vDash \phi), \\
 & & \text{where } V'(x) = a \text{ and } V'(y) = V(y) \text{ for } y \neq x; \\
M, V, w \vDash \exists x.\phi & \text{iff} & \text{there exists } a \in D \text{ such that } M, V', w \vDash \phi, \\
 & & \text{where } V'(x) = a \text{ and } V'(y) = V(y) \text{ for } y \neq x.
\end{array}
$$

If $M, V, w \vDash \phi$ then we say that $\phi$ is true at $w$ in $M$ w.r.t. $V$. We write $M, w \vDash \phi$ to denote that $M, V, w \vDash \phi$ for every $V$. We say that $M$ satisfies $\phi$, or $\phi$ is true in $M$, and write $M \vDash \phi$, if $M, \tau \vDash \phi$. For a set $\Gamma$ of formulas, we call $M$ a model of $\Gamma$ and write $M \vDash \Gamma$ if $M \vDash \alpha$ for every $\alpha \in \Gamma$.

A *logic* can be defined by a set of well-formed formulas, a class of admissible interpretations, and a satisfaction relation. The class of admissible interpretations for a modal logic $L$ is often specified by restrictions on Kripke frames admissible for $L$. We refer to such restrictions by *$L$-frame restrictions* and call frames with such properties *$L$-frames*.

**Definition 2.8** We call a model $M$ with $L$-frame an *$L$-model*. We say that $\phi$ is *$L$-satisfiable* if there exists an $L$-model of $\phi$, i.e. an $L$-model satisfying $\phi$. A formula $\phi$ is said to be *$L$-valid* and called an *$L$-tautology* if $\phi$ is true in every $L$-model. For a set $\Gamma$ of formulas, we write $\Gamma \vDash_L \phi$ and call $\phi$ a *logical consequence* of $\Gamma$ in $L$ if $\phi$ is true in every $L$-model of $\Gamma$.

If as the class of admissible interpretations we take the class of all Kripke models (with no restrictions on the accessibility relations) then we obtain the quantified multimodal logic $K_{(m)}$. This logic is axiomatized by the following system:

- axioms for the classical predicate logic (without identity)
- the $K$-axioms: $\Box_i(\phi \to \psi) \to (\Box_i\phi \to \Box_i\psi)$
- the Barcan formula axioms: $\forall x.\Box_i\phi \to \Box_i\forall x.\phi$
- the axioms defining $\Diamond_i$ : $\Diamond_i\phi \equiv \neg\Box_i\neg\phi$
- the modus ponens rule: $\dfrac{\phi \quad \phi \to \psi}{\psi}$
- the generalization rule: $\dfrac{\phi}{\forall x.\phi}$
- and the modal generalization rules: $\dfrac{\phi}{\Box_i\phi}$

Note that the converse Barcan formula $\Box_i\forall x.\phi \to \forall x.\Box_i\phi$ is a consequence of this axiomatization system. Every logic whose axiomatization is an extension of the system $K_{(m)}$ is called a *normal multimodal logic*. For further introduction to modal logics, we refer the reader to the books [11, 17, 9].

## 2.2 A Class of Basic Serial Multimodal Logics

In general, a normal multimodal logic can be characterized by axioms that are added to the systems $K_{(m)}$. Let's consider a class, denoted by $BSMM$, of basic serial multimodal logics. A $BSMM$ logic is a normal multimodal logic parameterized by relations $AD/1$, $AT/1$, $AI/2$, $AB/2$, $A4/3$, $A5/3$ on the set $\{1,\ldots,m\}$, where the numbers on the right are arities and $AD$ is required to be full. These relations specify the following axioms:

$$\begin{array}{ll}
\Box_i\phi \to \Diamond_i\phi & \text{if } AD(i) \\
\Box_i\phi \to \phi & \text{if } AT(i) \\
\Box_i\phi \to \Box_j\phi & \text{if } AI(i,j) \\
\phi \to \Box_i\Diamond_j\phi & \text{if } AB(i,j) \\
\Box_i\phi \to \Box_j\Box_k\phi & \text{if } A4(i,j,k) \\
\Diamond_i\phi \to \Box_j\Diamond_k\phi & \text{if } A5(i,j,k)
\end{array}$$

It can be shown that the above axioms correspond to certain properties (listed below) of the accessibility relations in the sense that by adding some of them to the system $K_{(m)}$ we obtain an axiomatization system which is sound and complete with respect to the class of admissible interpretations that satisfy the corresponding restrictions on the accessibility relations.

| Axiom | Corresponding Condition |
|---|---|
| $\Box_i\phi \to \Diamond_i\phi$ | $\forall u\ \exists v\ R_i(u,v)$ |
| $\Box_i\phi \to \phi$ | $\forall u\ R_i(u,u)$ |
| $\Box_i\phi \to \Box_j\phi$ | $R_j \subseteq R_i$ |
| $\phi \to \Box_i\Diamond_j\phi$ | $\forall u,v\ (R_i(u,v) \to R_j(v,u))$ |
| $\Box_i\phi \to \Box_j\Box_k\phi$ | $\forall u,v,w\ (R_j(u,v) \wedge R_k(v,w) \to R_i(u,w))$ |
| $\Diamond_i\phi \to \Box_j\Diamond_k\phi$ | $\forall u,v,w\ (R_i(u,v) \wedge R_j(u,w) \to R_k(w,v))$ |

For a $BSMM$ logic $L$, we define the set of $L$-frame restrictions to be the set of the restrictions corresponding to the tuples of the relations $AD$, $AT$, $AI$, $AB$, $A4$, $A5$. Then the axiomatization system $L$ is sound and complete with respect to the class of $L$-models. For a survey on axiomatization systems for first-order modal logics and methods for proving their soundness and completeness, we refer the reader to Garson's work [18].

We will sometimes use $BSMM$ to denote an arbitrary logic belonging to the $BSMM$ class.

## 2.3 Multimodal Logics about Belief

To reflect properties of belief, one can extend the system $K_{(m)}$ with some of the following axioms:

| Name | Schema | Meaning |
|---|---|---|
| (D) | $\Box_i\phi \to \neg\Box_i\neg\phi$ | belief is consistent |
| (I) | $\Box_i\phi \to \Box_j\phi$ if $i > j$ | subscript indicates degree of belief |
| (4) | $\Box_i\phi \to \Box_i\Box_i\phi$ | belief satisfies positive introspection |
| ($4_s$) | $\Box_i\phi \to \Box_j\Box_i\phi$ | belief satisfies strong positive introspection |
| (5) | $\neg\Box_i\phi \to \Box_i\neg\Box_i\phi$ | belief satisfies negative introspection |
| ($5_s$) | $\neg\Box_i\phi \to \Box_j\neg\Box_i\phi$ | belief satisfies strong negative introspection |

The following logic systems are intended for reasoning about multi-degree belief:

$$\begin{array}{rcl}
KDI4_s & = & K_{(m)} + (D) + (I) + (4_s) \\
KDI4 & = & K_{(m)} + (D) + (I) + (4) \\
KDI4_s5 & = & K_{(m)} + (D) + (I) + (4_s) + (5) \\
KDI45 & = & K_{(m)} + (D) + (I) + (4) + (5)
\end{array}$$

The axiom (I) gives $\Box_i\phi$ the meaning "$\phi$ is believed with degree $i$". It can be written as $\Diamond_j\phi \to \Diamond_i\phi$ if $i > j$. If we read $\Diamond_i\phi$ as "it is possible at degree $i$ that $\phi$", then the higher is degree, the weaker is possibility. A solution for this problem is reading $\Diamond_i\phi$ as "it is possible weakly at degree $i$ that

$\phi$". The axioms (5) are controversial as they are quite strong. For this reason, we study also $KDI4$ and $KDI4_s$. Note that the axiom $(5_s)$ is true in $KDI4_s5$.

For multi-agent systems, subscripts beside $\Box$ and $\Diamond$ stand for agents. For distributed systems of belief we can use the logic system

$$KD4_s5_s = K_{(m)} + (D) + (4_s) + (5_s)$$

In this system, agents have full access to belief bases of each other. They are members of a united system and viewed as "friends". In another kind of multi-agent systems, agents are "opponents" and they play against each other. Each agent tries to simulate epistemic states of others. To program for an agent one may need to use modal operators of other agents. One of suitable logics for this problem is:

$$KD45_{(m)} = K_{(m)} + (D) + (4) + (5)$$

To capture common belief of a group of agents, one can extend the logic $KD45_{(m)}$ with modal operators for groups of agents and some additional axioms. Suppose that there are $n$ agents and $m = 2^n - 1$. Let $g$ be an one-to-one function that maps every natural number less than or equal to $m$ to a nonempty subset of $\{1, \ldots, n\}$. Suppose that an index $1 \leq i \leq m$ stands for the group of agents whose indexes form the set $g(i)$. We can adopt the axioms $(D)$, (4), and additionally $(I_g) : \Box_i \phi \to \Box_j \phi$ if $g(i) \supset g(j)$ (i.e. $i$ indicates a group that contains the group identified by $j$), and $(5_a) : \neg\Box_i \phi \to \Box_i \neg\Box_i \phi$ if $g(i)$ is singleton (i.e. $i$ stands for an agent). Thus, for reasoning about belief and common belief, we can use the following system:

$$KD4I_g5_a = K_{(m)} + (D) + (4) + (I_g) + (5_a)$$

Note that all of the logics mentioned in this subsection belong to the $BSMM$ class. As specific cases of the $BSMM$ class, their axioms correspond to the following properties of the accessibility relations:

| Axiom | Corresponding Condition |
|---|---|
| (D) | $\forall u\ \exists v\ R_i(u, v)$ |
| (I) | $R_j \subseteq R_i$ if $i > j$ |
| $(I_g)$ | $R_j \subseteq R_i$ if $g(i) \supseteq g(j)$ |
| (4) | $\forall u, v, w\ (R_i(u, v) \land R_i(v, w) \to R_i(u, w))$ |
| $(4_s)$ | $\forall u, v, w\ (R_j(u, v) \land R_i(v, w) \to R_i(u, w))$ |
| (5) | $\forall u, v, w\ (R_i(u, v) \land R_i(u, w) \to R_i(w, v))$ |
| $(5_s)$ | $\forall u, v, w\ (R_j(u, v) \land R_i(u, w) \to R_i(v, w))$ |
| $(5_a)$ | as for (5) if $g(i)$ is singleton |

## 2.4 Ordering Kripke Models

A formula is in the *negative normal form* if it does not contain the connective $\to$, and each of its negations occurs immediately before a classical atom. Every formula can be transformed to the equivalent negative normal form in the usual way.

**Definition 2.9** A formula is called *positive* if its negative normal form does not contain negation. A formula is called *negative* if its negation is a positive formula.

**Definition 2.10** A model $M$ is said to be *less than* or *equal to* $N$, write $M \leq N$, if for any positive ground formula $\phi$, if $M$ satisfies $\phi$ then $N$ also satisfies $\phi$.

The relation $\leq$ in the above definition is a pre-order[2].

**Definition 2.11** Let $M = \langle D, W, \tau, R_1, \ldots, R_m, \pi \rangle$ and $N = \langle D', W', \tau', R'_1, \ldots, R'_m, \pi' \rangle$ be Kripke models. We say that $M$ is *less than or equal to $N$ w.r.t. a binary relation $r \subseteq W \times W'$*, and write $M \leq_r N$, if the following conditions hold:

[2] i.e. a reflexive and transitive binary relation

1. $r(\tau, \tau')$.

2. $\forall x, x', y \;\; R_i(x, y) \wedge r(x, x') \rightarrow \exists y' \;\; R'_i(x', y') \wedge r(y, y')$, for all $1 \leq i \leq m$.

3. $\forall x, x', y' \;\; R'_i(x', y') \wedge r(x, x') \rightarrow \exists y \; R_i(x, y) \wedge r(y, y')$, for all $1 \leq i \leq m$.

4. For any $x \in W$, $x' \in W'$ such that $r(x, x')$, and for any ground classical atom $E$, if $M, x \vDash E$ then $N, x' \vDash E$.

In the above definition, the first three conditions state that $r$ is a bisimulation of the frames of $M$ and $N$. Intuitively, $r(x, x')$ states that the world $x$ is less than or equal to $x'$.

**Lemma 2.1** *If $M \leq_r N$ then $M \leq N$.*

This lemma can be proved by induction on the length of $\phi$ that, if $\phi$ is a ground formula and $M, w \vDash \phi$ then $N, w \vDash \phi$.

### 2.5 Notations, Labeled Modal Operators, and Unification

Throughout this work, we will use the following notations:

- $\top$ – the *truth* symbol, with the usual[3] semantics;
- $D$, $E$, $F$ – classical atoms or $\top$;
- $X$, $Y$ – variables for classical atoms or $\top$, called *atom variables*;
- $\langle E \rangle_i$ – $\Diamond_i$ labeled by $E$;
- $\langle X \rangle_i$ – $\Diamond_i$ labeled by $X$;
- $\nabla$ – $\Box_i$, $\Diamond_i$, $\langle E \rangle_i$, or $\langle X \rangle_i$, called a modal operator;
- $\triangle$ – a sequence of modal operators, which can be empty and is called a *modality*;
- $A$, $B$ – formulas of the form $E$ or $\nabla E$, called *simple atoms*;
- $\alpha$, $\beta$ – formulas of the form $\triangle E$, called *atoms*;
- $\phi$, $\psi$ – *labeled formulas* (i.e. formulas that may contain $\langle E \rangle_i$ and $\langle X \rangle_i$), which will be simply called formulas.

A *ground formula* is redefined to be a formula with no variables and no atom variables. A *closed formula* is redefined to be a formula with no atom variables and no free occurrences of variables. A modal operator is said to be *ground* if it is $\Box_i$, $\Diamond_i$, or $\langle E \rangle_i$ with $E$ being $\top$ or a ground classical atom. A *ground modality* is a modality that contains only ground modal operators. A *labeled modal operator* is a modal operator of the form $\langle E \rangle_i$ or $\langle X \rangle_i$. A modality is said to be *universal* if it contains only universal modal operators.

Recall that a simple subscript like $i$ beside $\Box$, $\Diamond$, $\langle E \rangle$, or $\langle X \rangle$ indicates the *kind* (i.e. degree/agent number) of the modal operator. We use such subscripts beside $\nabla$ for the same aim. To distinguish a number of modal operators we use superscripts of the form $(i)$, e.g. $\Box^{(1)}$, $\Box^{(2)}$, $\nabla^{(i)}$, $\nabla^{(i')}$.

Denote $EdgeLabels = \{\langle E \rangle_i \mid E \in \mathcal{B} \cup \{\top\} \text{ and } 1 \leq i \leq m\}$. The semantics of $\langle E \rangle_i$ for $E$ being $\top$ or a ground classical atom is specified by the following definition.

**Definition 2.12** Let $M = \langle D, W, \tau, R_1, \ldots, R_m, \pi \rangle$ be a Kripke model. A *$\Diamond$-realization function on $M$* is a partial function $\sigma : W \times EdgeLabels \rightarrow W$ such that if $\sigma(w, \langle E \rangle_i) = u$, then $R_i(w, u)$ holds and $M, u \vDash E$. Given a $\Diamond$-realization function $\sigma$, a world $w \in W$, and a ground formula $\phi$, the satisfaction relation $M, \sigma, w \vDash \phi$ is defined in the usual way, except that $M, \sigma, w \vDash \langle E \rangle_i \psi$ iff $\sigma(w, \langle E \rangle_i)$ is defined and $M, \sigma, \sigma(w, \langle E \rangle_i) \vDash \psi$. We write $M, \sigma \vDash \phi$ to denote that $M, \sigma, \tau \vDash \phi$. For a set $I$ of ground atoms, we write $M, \sigma \vDash I$ to denote that $M, \sigma \vDash \alpha$ for all $\alpha \in I$; we write $M \vDash I$ and call $M$ a model of $I$ if $M, \sigma \vDash I$ for *some* $\sigma$.

[3] i.e. it is always true that $M, V, w \vDash \top$

**Definition 2.13** Let $\sigma$ and $\sigma'$ be $\Diamond$-realization functions on a model $M$. We say that $\sigma$ is an extension of $\sigma'$ if whenever $\sigma'(w, \langle E\rangle_i)$ is defined then $\sigma(w, \langle E\rangle_i) = \sigma'(w, \langle E\rangle_i)$.

**Definition 2.14** A $\Diamond$-realization function $\sigma$ on $M$ is said to be *maximal* if $\sigma(w, \langle E\rangle_i)$ is defined whenever $M, w \vDash \Diamond_i E$.

We now give definitions for substitution and unification.

**Definition 2.15** A *substitution* $\theta$ is a (finite or infinite) set of the form $\{x_1/t_1, x_2/t_2, \ldots, X_1/E_1, X_2/E_2, \ldots, Y_1/Z_1, Y_2/Z_2, \ldots\}$, where $x_1, x_2, \ldots$ are distinct variables, $t_1, t_2, \ldots$ are terms, $X_1, X_2, \ldots, Y_1, Y_2, \ldots$ are distinct atom variables, and for any element $v/s$ of the set, $s$ is distinct from $v$. The set $\{x_1, x_2, \ldots, X_1, X_2, \ldots, Y_1, Y_2, \ldots\}$ is called the *domain* of $\theta$ and denoted by $Dom(\theta)$. $\theta$ is called a *ground substitution* if the set $\{Y_1, Y_2, \ldots\}$ is empty, $t_1, t_2, \ldots$ are ground terms, and $E_1, E_2, \ldots$ are ground classical atoms. The substitution given by the empty set is called the *identity substitution* and denoted by $\varepsilon$.

An *expression* is either a term or a formula. Let $\theta = \{v_1/s_1, v_2/s_2, \ldots\}$ be a substitution and $\Phi$ an expression. Then $\Phi\theta$, the *instance* of $\Phi$ by $\theta$, is the expression obtained from $\Phi$ by simultaneously replacing each occurrence of $v_i$ in $\Phi$ by $s_i$. For example, if $\Phi = p(x, y, a)$ and $\theta = \{x/y,\ y/f(x, b)\}$, then $\Phi\theta = p(y, f(x, b), a)$. If $S$ is a set of expressions, then by $S\theta$ we denote the set $\{\Phi\theta \mid \Phi \in S\}$.

Let $\theta$ and $\gamma$ be substitutions. Let $\delta = \{v/(s\gamma) \mid v/s \in \theta\} \cup \{v/s \mid v/s \in \gamma \text{ and } v \notin Dom(\theta)\}$. Then the *composition* $\theta\gamma$ is the substitution obtained from $\delta$ by deleting every element $v/s$ with $s = v$. Here are some properties of substitutions: if $\theta, \gamma, \delta$ are substitutions and $\Phi$ is an expression, then $\theta\varepsilon = \varepsilon\theta = \theta$, $(\Phi\theta)\gamma = \Phi(\theta\gamma)$, and $(\theta\gamma)\delta = \theta(\gamma\delta)$.

**Definition 2.16** Let $\Phi$ and $\Psi$ be expressions. We say that $\Phi$ is a *variant* of $\Psi$ if there exist substitutions $\theta$ and $\gamma$ such that $\Phi = \Psi\theta$ and $\Psi = \Phi\gamma$.

**Definition 2.17** Let $S$ be a finite set of expressions. A substitution $\theta$ is called a *unifier* for $S$ if $S\theta$ is singleton. A unifier $\theta$ for $S$ is called a *most general unifier* (mgu) for $S$ if, for each unifier $\gamma$ of $S$, there exists a substitution $\delta$ such that $\gamma = \theta\delta$. If there exists a unifier for $S$ then $S$ is said to be *unifiable*.

The well-known unification theorem says that if $S$ is a finite unifiable set of expressions, then there exists an mgu for $S$, which can be effectively computed (see, e.g., [22]).

Atom variables in modal operators of the form $\langle X\rangle_i$ are mainly interpreted by substitutions. When a formula $\phi$ is taken to be semantically considered, all modal operators $\langle X\rangle_i$ in $\phi$ are treated as[4] $\langle\top\rangle_i$, which is formalized by the following definition.

**Definition 2.18** Given a Kripke model $M$, a $\Diamond$-realization function $\sigma$, and a labeled formula $\phi$ without quantifiers, we write $M, \sigma \vDash \forall_c(\phi)$ to denote that for any substitution $\theta$ which substitutes every variable by a ground term and does not substitute atom variables, $M, \sigma \vDash \phi\,\theta\,\delta_\top$, where $\delta_\top = \{X/\top \mid X \text{ is an atom variable}\}$. By $M \vDash \forall_c(\phi)$ we denote $M, \sigma \vDash \forall_c(\phi)$ for *some* $\sigma$. For a formula set $\Gamma$ and a formula $\phi$ (maybe in the form $\forall_c(\psi)$), we write $\Gamma \vDash_L \phi$ to denote that for any $L$-model $M$, if $M \vDash \Gamma$ then $M \vDash \phi$.

The quantifier $\forall_c$ is introduced because $\Diamond$-realization functions are defined using Herbrand base and we do not want to restrict only to Herbrand models. Suppose that there are enough constant symbols, for example, infinitely many. Then, because a $BSMM$ logic $L$ has a complete axiomatization, for $\Gamma$ being a finite formula set and $\phi$ a formula – both without labeled modal operators, $\Gamma \vDash_L \forall(\phi)$ iff $\Gamma \vDash_L \forall_c(\phi)$.

[4]Atom variables appear only in *goal* bodies (see Definition 3.4). In the negation of a goal (i.e. a query) they are existentially quantified. Hence it is sufficient to choose some concrete values for them. Furthermore, as we will see, the modal operator $\langle\top\rangle_i$ plays the role of $\Box_i$; and if $X$ remains at the end as an unsubstituted atom variable then $\langle X\rangle_i$ intuitively also plays the role of $\Box_i$.

## 3 Positive Multimodal Logic Programs

In [28] we presented a logic programming language called MProlog for monomodal logics. In this section, we extend this language for multimodal logics, using the same name for the new one. The defined language is as expressive as the general Horn fragment in the considered multimodal logics. For $L$ being one of the multimodal logics about belief, we adopt some restrictions on MProlog to obtain $L$-MProlog. The restrictions do not reduce the expressiveness of the language and are acceptable from the practical point of view. This section is free from labeled modal operators, hereby we assume that formulas in this section do not contain labeled modal operators.

In the clausal notation, we write $\triangle(\phi \leftarrow \psi_1, \ldots, \psi_n)$ to denote the formula $\forall(\triangle(\phi \vee \neg\psi_1 \vee \ldots \vee \neg\psi_n))$, where $\phi$, $\psi_1$, ..., $\psi_n$ are positive formulas not containing quantifiers.

**Definition 3.1** A *program clause* is a formula of the form

$$\triangle(A \leftarrow B_1, \ldots, B_n)$$

where $\triangle$ is a (possibly empty) sequence of universal modal operators, $n \geq 0$, and $A$, $B_1$, ..., $B_n$ are formulas of the form $E$, $\Box_i E$, or $\Diamond_i E$ with $E$ being a classical atom. $\triangle$ is called the *modal context*, $A$ the *head*, and $B_1, \ldots, B_n$ the *body* of the program clause.

**Definition 3.2** The clause with empty head and empty body is called the *empty clause* and denoted by $\diamond$.

**Definition 3.3** An *MProlog program* is a finite set of program clauses.

**Definition 3.4** An *MProlog goal atom* is a formula of the form $\triangle E$ or $\triangle\Diamond_i E$, where $\triangle$ is a (possibly empty) sequence of universal modal operators. An *MProlog query* is a formula of the form $\exists(\alpha_1 \wedge \ldots \wedge \alpha_k)$, where $\alpha_1$, ..., $\alpha_k$ are MProlog goal atoms. An *MProlog goal* is the negation of an MProlog query, written in the form $\leftarrow \alpha_1, \ldots, \alpha_k$.

If $P$ is an MProlog program, $Q = \exists(\alpha_1 \wedge \ldots \wedge \alpha_k)$ is an MProlog query and $G = \leftarrow \alpha_1, \ldots, \alpha_k$ is the corresponding goal, then $P \vDash_L Q$ iff $P \cup \{G\}$ is $L$-unsatisfiable. For the proof of this statement, just note that $G = \forall(\neg(\alpha_1 \wedge \ldots \wedge \alpha_k))$.

When the base logic is intended for reasoning about multi-degree belief, it has little sense to write a program clause in the form $\Box_i\Box_j\phi$ or a goal in the form $\leftarrow \Box_i\Box_j\alpha$ or $\leftarrow \Box_i\Diamond_j\alpha$. Besides, in the logics $KDI4_s5$ and $KD4_s5_s$ we have the tautology $\nabla^{(1)}\nabla^{(2)}\phi \equiv \nabla^{(2)}\phi$. For these reasons, we introduce some restrictions for MProlog programs and goals in these logics.

**Definition 3.5** For $L \in \{KDI4_s, KDI4, KDI4_s5, KDI45, KD4_s5_s\}$, an MProlog program is called an *$L$-MProlog program* if its program clauses have modal context with length bounded by 1, an MProlog goal is called an *$L$-MProlog goal* if its goal atoms are in the form $\triangle E$ with $|\triangle| \leq 1$.

In the logic $KD45_{(m)}$, we have the tautologies $\Box_i\Box_i\phi \equiv \Box_i\phi$ and $\Box_i\Diamond_i\phi \equiv \Diamond_i\phi$. In $KD4I_g5_a$, these two equivalences hold for the case when $g(i)$ is singleton. So, we introduce some restrictions for MProlog programs and goals in $KD45_{(m)}$ and $KD4I_g5_a$.

**Definition 3.6** An MProlog program is called a *$KD45_{(m)}$-MProlog program* if the modal contexts of its program clauses do not contain subsequences of the form $\Box_i\Box_i$. An MProlog goal is called a *$KD45_{(m)}$-MProlog goal* if each of its goal atoms $\triangle E$ satisfies the condition that $\triangle$ does not contain subsequences of the form $\Box_i\Box_i$ or $\Box_i\Diamond_i$. *$KD4I_g5_a$-MProlog* programs and goals are defined similarly with the condition that $g(i)$ is singleton.

For $L$ not mentioned in the two above definitions, no restrictions on the form of $L$-MProlog programs and goals are adopted.

**Definition 3.7** An *($L$-)MProlog-$\Box$ program* (resp. *goal*) is an ($L$-)MProlog program (resp. goal) that does not contain existential modal operators $\Diamond_i$.

An extension of MProlog, called *eMProlog*, is defined in the same way as in [28] and stands for the general Horn fragment.

**Definition 3.8** A formula $\phi$ without quantifiers is called a *non-negative modal Horn formula (without quantifiers)* if one of the following conditions holds:

- $\phi$ is a classical atom;
- $\phi = \psi \leftarrow \zeta$, where $\psi$ is a non-negative modal Horn formula and $\zeta$ is a positive formula (in the negative normal form);
- $\phi = \Box_i\psi$, or $\phi = \Diamond_i\psi$, or $\phi = \psi \wedge \zeta$, where $\psi$ and $\zeta$ are non-negative modal Horn formulas.

**Definition 3.9** An *eMProlog program* is a finite set of formulas of the form $\forall(\phi)$, where $\phi$ is a non-negative modal Horn formula without quantifiers. An *eMProlog query* is a formula of the form $\exists(\phi)$, where $\phi$ is a positive formula without quantifiers. An *eMProlog goal* is the negation of an eMProlog query.

We define below answers and correct answers.

**Definition 3.10** Let $P$ be an MProlog (resp. eMProlog) program and $G$ an MProlog (resp. eMProlog) goal. An *answer* $\theta$ for $P \cup \{G\}$ is a substitution for variables of $G$ (i.e. if $x_1, \ldots, x_n$ are all variables of $G$, then $\theta = \{x_{i_1}/t_1, \ldots, x_{i_k}/t_k\}$ for some $1 \leq i_1 < \ldots < i_k \leq n$ and some terms $t_1, \ldots, t_k$).

**Definition 3.11** Let $L$ be a multimodal logic, $P$ an MProlog (resp. eMProlog) program, $Q = \exists(\phi)$ an MProlog (resp. eMProlog) query and $G$ the corresponding goal (i.e. $G = \neg Q$). Let $\theta$ be an answer for $P \cup \{G\}$. We say that $\theta$ is a *correct answer* in $L$ for $P \cup \{G\}$ if $P \vDash_L \forall(\phi\theta)$.

The following proposition states that MProlog and $L$-MProlog, where $L$ is one of the considered multimodal logics, have the same expressiveness as eMProlog.

**Proposition 3.1** *Let $L$ be a BSMM logic. For any eMProlog program $P$ and any eMProlog goal $G$, there exist an MProlog program $P'$ and an MProlog goal $G'$ such that:*

- *Every correct answer in $L$ for $P \cup \{G\}$ is a correct answer in $L$ for $P' \cup \{G'\}$ and vice versa.*
- *If $L \in \{KDI4_s, KDI4, KDI4_s5, KDI45, KD4_s5_s, KD45_{(m)}, KD4I_g5_a\}$, then $P'$ is an $L$-MProlog program and $G'$ is an $L$-MProlog goal.*
- *$P'$ and $G'$ can be obtained from $P$ and $G$ in polynomial time.*

See Appendix A for the proof of this proposition.

# 4 A Framework for Multimodal Logic Programming

As standard semantics for classical definite logic programs there are the least model semantics, the fixpoint semantics and the SLD-resolution calculus (a procedural semantics). See Minker's work [24] for a survey and the works by Lloyd [22] and Apt [3] for foundations of classical logic programming. In this section, we give a general framework for developing such mentioned semantics for $L$-MProlog programs. The base logic $L$ is required so that the set of $L$-frame restrictions consists of $\forall x\ \exists y\ R_i(x,y)$ (seriality), for $1 \leq i \leq m$, and some classical first-order Horn clauses. We prove the main results using certain lemmas and theorems, which are really dependent on $L$ and left as "to be proved". For a specific logic $L$, lemmas and theorems with that remark need to be proved to guarantee correctness of the main theorems w.r.t. that logic.

## 4.1 Model Generators

In classical logic programming, the direct consequences operator $T_P$ acts on sets of ground atoms. It computes "direct" consequences of the input set using the program clauses of $P$. The operator is monotonic and continuous and has the least fixpoint, which is a set of atoms and forms the least model of $P$. In modal logic programming, to obtain a similar result we first have to decide what is the domain of the direct consequences operator $T_{L,P}$. Naturally, we still want it to be the class of sets of *atoms*. But what is an *atom* in this case? When applying $T_{L,P}$, if we obtain some atom of the form $\triangle\Diamond_i E$, then to simplify the task we want to label the modal operator $\Diamond_i$. Labeling allows us to address the chosen world(s) in which this particular $E$ must hold. A natural way is to label $\Diamond_i$ by $E$ to obtain $\langle E\rangle_i$. Thus, an output/input of $T_{L,P}$ consists of atoms of the form $\triangle E$, where $\triangle$ is a sequence of modal operators of the form $\Box_i$ or $\langle F\rangle_i$, with $E$, $F$ being ground classical atoms. For the reason that the least fixpoint of $T_{L,P}$ should represent a least $L$-model of $P$, we call inputs/outputs of $T_{L,P}$ *model generators*.

**Definition 4.1** A *model generator* is a set of ground atoms not containing $\Diamond_i$, $\langle\top\rangle_i$, $\top$.

Because an atom in $L$ may be reducible to some more compact form, for each specific $L$ we will define the *$L$-normal form of modalities*. It is possible that there are no restrictions on the $L$-normal form of modalities.

**Definition 4.2** A modality is in the *$L$-normal labeled form* if it is in the $L$-normal form and does not contain modal operators of the form $\Diamond_i$ or $\langle\top\rangle_i$. An atom is in the *$L$-normal (labeled) form* if it is of the form $\triangle E$ with $\triangle$ in the $L$-normal (labeled) form. An atom is in the *almost $L$-normal labeled form* if it is of the form $\triangle A$ with $\triangle$ in the $L$-normal labeled form.

**Example 4.1** Define that a modality is in the $KDI4_s5$-normal form if its length is 0 or 1. Then the modalities $\Box_i$ and $\langle E\rangle_i$ are in the $KDI4_s5$-normal labeled form, while $\Box_i\Box_j$, $\Diamond_i$ and $\langle\top\rangle_i$ are not. Atoms $E$, $\Box_i E$ and $\langle E\rangle_i F$ are in the $KDI4_s5$-normal labeled form, while $\Box_i\Box_j E$, $\Diamond_i E$ and $\langle\top\rangle_i E$ are not. Atoms $E$, $\Box_i E$, $\Diamond_i E$, $\Box_i\Box_j E$, $\Box_i\Diamond_j E$ and $\langle F\rangle_i E$ are in the almost $KDI4_s5$-normal labeled form, while $\Diamond_i\Box_j E$ and $\Box_i\Box_j\Box_k E$ are not.

**Definition 4.3** An *$L$-normal model generator* is a set of ground atoms in the $L$-normal labeled form.

An $L$-normal model generator $I$ is expected to represent an $L$-model. This specific model is called the *standard $L$-model* of $I$. It should contain only (positive) information that come from $I$. This means that the standard $L$-model of $I$ should be a least $L$-model of $I$.

Given an $L$-normal model generator $I$, we can construct a least $L$-model for it by building an $L$-model graph realizing $I$ (cf. [26]). Formulas of the form $\Box_i\alpha$ are realized in the usual way; a formula of the form $\langle E\rangle_i\alpha$ is realized at a world $w$ by connecting $w$ to a world identified by $w\langle E\rangle_i$ via $R_i$ and adding $\alpha$ to that world. To guarantee the constructed model graph to be the smallest, each new world is connected via each $R_i$ to an empty world at the time of its creation. Sometimes, the accessibility relations are extended to satisfy all of the $L$-frame restrictions. We want, however, to give a more declarative definition of the standard $L$-model of an $L$-normal model generator $I$. The part specific to $L$ is extracted into $Ext_L$ and $Serial_L$, where $Ext_L(I)$ is an $L$-normal model generator that is a superset of $I$, and $Serial_L$ is a set of atoms of the form $\triangle\langle\top\rangle_i\top$ with $\triangle$ being a universal modality. The standard $L$-model of $I$ is then defined using $Ext_L(I)$ and $Serial_L$ in a unified way, independently of $L$. The set $Serial_L$ is intended to guarantee that, for every world $w$ and $1 \leq i \leq m$, $w$ will be connected to a world which is "less than or equal to" every world accessible from $w$ via $R_i$.

**Definition 4.4** Define $Serial_L = \{\triangle\langle\top\rangle_i\top \mid 1 \leq i \leq m$ and $\triangle$ is a universal modality such that $\triangle\langle\top\rangle_i$ is in the $L$-normal form$\}$.

A *forward rule* is a schema of the form $\alpha \to \beta$, while a *backward rule* is a schema of the form $\alpha \leftarrow \beta$. A rule can be accompanied with some conditions specifying when the rule can be applied. We use forward rules to specify the operators $Ext_L$ and $Sat_L$ (needed for defining fixpoint semantics) and use backward rules as meta-clauses when dealing with SLD-resolution calculi. In practice, conditions for applying a backward rule can be attached to the body of the rule, and in general, a backward rule can be of the form $(\alpha \leftarrow \phi, \beta, \psi)$ with $\phi$ and $\psi$ being conjunctions of classical atoms. In this work, we just take the definition that a backward rule is of the form $\alpha \leftarrow \beta$.

**Definition 4.5** The *operator* $Ext_L$ is specified by a finite set of forward rules. Given an $L$-normal model generator $I$, $Ext_L(I)$ is the least extension of $I$ that contains all ground atoms in the $L$-normal labeled form that are derivable from some atom of $I$ using the rules specifying $Ext_L$.

Note that $Ext_L(I)$ is an $L$-normal model generator if so is $I$.

**Definition 4.6** Let $I$ be an $L$-normal model generator. The *standard $L$-model* of $I$ is defined as follows. Let $W' = EdgeLabels^*$ (i.e. the set of finite sequences of elements of $\{\langle E\rangle_i \mid E \in \mathcal{B} \cup \{\top\}$ and $1 \leq i \leq m\}$), $\tau = \epsilon$, $H(\tau) = Ext_L(I) \cup Serial_L$. Let $R'_i \subseteq W' \times W'$ and $H(u)$, for $u \in W'$, $u \neq \tau$, be the least sets such that:

- if $\langle E\rangle_i\alpha \in H(w)$, then $R'_i(w, w\langle E\rangle_i)$ and $\{E, \alpha\} \subseteq H(w\langle E\rangle_i)$;
- if $\Box_i\alpha \in H(w)$ and $R'_i(w, w\langle E\rangle_i)$, then $\alpha \in H(w\langle E\rangle_i)$.

Let $R_i$, for $1 \leq i \leq m$, be the least[5] extension of $R'_i$ such that $\{R_i \mid 1 \leq i \leq m\}$ satisfies all the restrictions of $L$-frames, not counting seriality (which is cared by $Serial_L$). Let $W$ be $W'$ without worlds not reachable directly nor indirectly from $\tau$ via the accessibility relations $R_i$. We call the model graph $\langle W, \tau, R_1, \ldots, R_m, H\rangle$ the *standard $L$-model graph* of $I$, and its corresponding model $M$ the *standard $L$-model* of $I$. $\{R'_i \mid 1 \leq i \leq m\}$ is called the *skeleton* of $M$. By the *standard $\Diamond$-realization function on $M$* we call the $\Diamond$-realization function $\sigma$ defined as follows: if $R'_i(w, w\langle E\rangle_i)$ then $\sigma(w, \langle E\rangle_i) = w\langle E\rangle_i$, else $\sigma(w, \langle E\rangle_i)$ is undefined.

**Definition 4.7** If a modality $\triangle$ is obtainable from $\triangle'$ by replacing some (possibly zero) $\nabla_i$ by $\Box_i$ then we call $\triangle$ a *$\Box$-lifting form* of $\triangle'$. If $\triangle$ is a $\Box$-lifting form of $\triangle'$ then we call an atom $\triangle\alpha$ a *$\Box$-lifting form* of $\triangle'\alpha$. For example, $\Box_1\Box_2\langle p(a)\rangle_1\Box_2 q(b)$ is a $\Box$-lifting form of $\langle X\rangle_1\langle p(c)\rangle_2\langle p(a)\rangle_1\Diamond_2 q(b)$.

The following lemma will be used to prove, among others, Lemma 4.2.

**Lemma 4.1** *Let $I$ be an $L$-normal model generator and $M = \langle W, \tau, R_1, \ldots, R_m, H\rangle$ the standard $L$-model graph of $I$. Let $w = \langle E_1\rangle_{i_1} \ldots \langle E_k\rangle_{i_k}$ be a world of $M$ and $\triangle = w$ be a modality. Then for $\alpha$ not containing $\top$, $\alpha \in H(w)$ iff there exists a $\Box$-lifting form $\triangle'$ of $\triangle$ such that $\triangle'\alpha \in Ext_L(I)$.*

This lemma can be proved by induction on the length of $w$ in a straightforward way.

The following lemma states that Definition 4.6 is admissible (i.e. the standard $L$-model of $I$ is really an $L$-model of $I$).

**Lemma 4.2 (to be proved)** *Let $I$ be an $L$-normal model generator, $M$ the standard $L$-model of $I$, and $\sigma$ the standard $\Diamond$-realization function on $M$. Then $M$ is an $L$-model and $M, \sigma \models I$.*

**Theorem 4.3 (to be proved)** *The standard $L$-model of an $L$-normal model generator $I$ is a least $L$-model of $I$.*

## 4.2 Fixpoint Semantics

We now return to the direct consequences operator $T_{L,P}$. Given an $L$-normal model generator $I$, how can $T_{L,P}(I)$ be defined? Basing on the axioms of $L$, $I$ is first extended to the *$L$-saturation* of $I$, denoted by $Sat_L(I)$, which is a set of atoms. Next, *$L$-instances of program clauses* of $P$ are *applied* to the atoms of $Sat_L(I)$. This is done by the operator $T_{0L,P}$. The set $T_{0L,P}(Sat_L(I))$ is a model

[5] the least extension exists due to the assumption that all $L$-frame restrictions not concerning seriality are classical first-order Horn formulas

generator but not necessary in the $L$-normal form. Finally, the *normalization operator* $NF_L$ converts $T_{0L,P}(Sat_L(I))$ to an $L$-normal model generator. $T_{L,P}(I)$ is defined as $NF_L(T_{0L,P}(Sat_L(I)))$.

We will define a pre-order $\preceq_L$ between modal operators for each specific logic $L$ to decide whether a given modality is an $L$-instance of another one. We require that $\Diamond_i \preceq_L \langle E\rangle_i \preceq_L \Box_i$, $\Diamond_i \preceq_L \langle X\rangle_i \preceq_L \Box_i$, and if $\nabla \preceq_L \langle E\rangle_i$ and $\nabla \neq \langle E\rangle_i$ then $\nabla \preceq_L \langle X\rangle_i$. As examples, we have the following definition.

**Definition 4.8** For $L$ being one of the considered multimodal logics, define $\preceq_L$ to be the least reflexive and transitive relation between modal operators such that:

- $\Diamond_i \preceq_L \langle E\rangle_i \preceq_L \Box_i$ and $\Diamond_i \preceq_L \langle X\rangle_i \preceq_L \Box_i$,
- $\Box_i \preceq_L \Box_j$ and $\Diamond_j \preceq_L \Diamond_i$ if $L \in \{KDI4_s, KDI4, KDI4_s5, KDI45\}$ and $i \leq j$,
- $\Box_i \preceq_L \Box_j$ and $\Diamond_j \preceq_L \Diamond_i$ if $L = KD4I_g5_a$ and $g(i) \subseteq g(j)$.

**Definition 4.9** An atom $\nabla^{(1)} \ldots \nabla^{(n)}\alpha$ is called an *$L$-instance* of an atom $\nabla^{(1')} \ldots \nabla^{(n')}\alpha'$ if there exists a substitution $\theta$ such that $\alpha = \alpha'\theta$ and, for $1 \leq i \leq n$, $\nabla^{(i)} \preceq_L \nabla^{(i')}\theta$ (treating $\nabla^{(i')}$ as an expression). A modality $\triangle$ is called an *$L$-instance* of $\triangle'$, and we also say that $\triangle'$ is *equal to* or *more general in L than* $\triangle$ (hereby we define a *pre-order between modalities*), if $\triangle E$ is an $L$-instance of $\triangle' E$ for some ground classical atom $E$.

The following lemma clearly holds for $\preceq_L$ defined as in Definition 4.8.

**Lemma 4.4 (to be proved)** *If a modality $\triangle$ is an $L$-instance of a universal modality $\triangle'$ then $\triangle'\phi \vDash_L \Box_{i_1} \ldots \Box_{i_h}\phi$, where $\Box_{i_1} \ldots \Box_{i_h}$ is a $\Box$-lifting form of $\triangle$ and $\phi$ is a classical formula.*

**Definition 4.10** Let $\triangle$ be a universal modality in the $L$-normal form and $\triangle'$ a modal context of an $L$-MProlog program clause. We say that $\triangle$ is an *$L$-context instance* of $\triangle'$ if $\triangle'\phi \to \triangle\phi$ is $L$-valid.

Observe that if the decision problem in the *propositional* version of $L$ is decidable then the problem of checking $L$-context instances is also decidable. For all of the multimodal logics about belief considered in this work, these two problems are decidable and the latter is much simpler[6].

**Definition 4.11** Let $\phi$ and $\phi'$ be program clauses with empty modal context, $\triangle$ a universal modality in the $L$-normal form, and $\triangle'$ a modal context of an $L$-MProlog program clause. We say that $\triangle\phi$ is an *$L$-instance* of (a program clause) $\triangle'\phi'$ if $\triangle$ is an $L$-context instance of $\triangle'$ and there exists a substitution $\theta$ such that $\phi = \phi'\theta$.

We now give definitions concerning $Sat_L$, $T_{0L,P}$, and $NF_L$.

**Definition 4.12** The *saturation operator* $Sat_L$ is specified by a finite set of forward rules. Given an $L$-normal model generator $I$, $Sat_L(I)$ is the least extension of $I$ that contains all ground atoms in the almost $L$-normal labeled form that are derivable from some atom in $I$ using the rules specifying $Sat_L$.

We expect that each rule specifying $Ext_L$ is also a rule specifying $Sat_L$, which implies $Ext_L(I) \subseteq Sat_L(I)$. We require the following property of $Sat_L$:

**Lemma 4.5 (to be proved)** *Let $I$ be an $L$-normal model generator, $M$ the standard $L$-model of $I$, and $\alpha$ a ground $L$-MProlog goal atom. Suppose that $M \vDash \alpha$. Then $\alpha$ is an $L$-instance of some atom of $Sat_L(I)$.*

[6]Let $\triangle$ and $\triangle'$ be as in Definition 4.10. For $L \in \{KDI4_s, KDI4, KDI4_s5, KDI45, KD4_s5_s, KD45_{(m)}\}$ and the $L$-normal form of modalities defined later in Tables 2, 3, 4, 5, 6, and 7, $\triangle$ is an $L$-context instance of $\triangle'$ iff $\triangle = \triangle'$ or one of the following condition holds:

- $L \in \{KDI4_s5, KD4_s5_s\}$ and $\triangle$ is an $L$-instance of $\triangle'$;
- $L = KDI4_s$, $\triangle' = \Box_i$, and the last modal operator of $\triangle$ is $\Box_j$ with $j \leq i$;
- $L \in \{KDI4, KDI45\}$, $\triangle' = \Box_i$, $\triangle$ is not empty, and every modal operator $\Box_j$ of $\triangle$ satisfies $j \leq i$.

**Definition 4.13** The *forward labeled form* of an atom $\alpha$ is the atom $\alpha'$ such that if $\alpha$ is of the form $\triangle\Diamond_i E$ then $\alpha' = \triangle\langle E\rangle_i E$, else $\alpha' = \alpha$.

**Definition 4.14** Let $P$ be an $L$-MProlog program. The *operator* $T_{0L,P}$ is defined as follows: for a set $I$ of ground atoms in the almost $L$-normal labeled form, $T_{0L,P}(I)$ is the least (w.r.t. $\subseteq$) model generator such that if $\triangle(A \leftarrow B_1, \ldots, B_n)$ is a ground $L$-instance of some program clause of $P$ and $\triangle'$ is a maximally general[7] ground modality in the $L$-normal labeled form such that $\triangle'$ is an $L$-instance of $\triangle$ and $\triangle' B_i$ is an $L$-instance of some atom of $I$ (for every $1 \leq i \leq n$), then the forward labeled form of $\triangle' A$ belongs to $T_{0L,P}(I)$.

For $I$ being an $L$-normal model generator, $T_{0L,P}(Sat_L(I))$ is a model generator, but not necessary in the $L$-normal form.

**Definition 4.15** The *normalization operator* $NF_L$ is specified by a finite set of forward rules. Given a model generator $I$, $NF_L(I)$ is the set of ground atoms in the $L$-normal labeled form that are derivable from some atom of $I$ using the rules specifying $NF_L$.

We expect that if $I$ is singleton then $NF_L(I)$ is also singleton. If there are no conditions on the $L$-normal form of atoms, then the set of rules specifying $NF_L$ is empty and $NF_L(I) = I$.

**Definition 4.16** Define $T_{L,P}(I) = NF_L(T_{0L,P}(Sat_L(I)))$.

By definition, $Sat_L$, $T_{0L,P}$, and $NF_L$ are all increasingly monotonic and compact. Hence so is $T_{L,P}$, and we have the following lemma.

**Lemma 4.6** *The operator $T_{L,P}$ is monotonic and continuous.*

Denote $T_{L,P}\uparrow 0 = \emptyset$, and $T_{L,P}\uparrow n = T_{L,P}(T_{L,P}\uparrow(n-1))$ for $n > 0$. For $P$ being an $L$-MProlog program, by the fixpoint theorem by Knaster and Tarski, the operator $T_{L,P}$ has the least fixpoint $T_{L,P}\uparrow\omega = \bigcup_{n=0}^{\omega} T_{L,P}\uparrow n$.

**Notation 4.1** Denote the least fixpoint $T_{L,P} \uparrow \omega$ by $I_{L,P}$ and the standard $L$-model of $I_{L,P}$ by $M_{L,P}$.

**Definition 4.17** Let $P$ be an $L$-MProlog program. An $L$-normal model generator $I$ is called an *$L$-model generator of $P$* if $T_{L,P}(I) \subseteq I$.

As a property of the least fixpoint, $I_{L,P}$ is the least (w.r.t. $\subseteq$) $L$-model generator of $P$.

**Lemma 4.7 (to be proved)** *For an L-MProlog program $P$, $P \models_L I_{L,P}$.*

**Lemma 4.8 (to be proved)** *Let $P$ be an L-MProlog program and $I$ an L-model generator of $P$. Then the standard L-model of $I$ is an L-model of $P$.*

Using these lemmas we can derive the following theorem.

**Theorem 4.9** *For an L-MProlog program $P$, $M_{L,P}$ is a least L-model of $P$.*

*Proof.* By Lemma 4.8, $M_{L,P}$ is an $L$-model of $P$. Let $M$ be an arbitrary $L$-model of $P$. By Lemma 4.7, $M \models I_{L,P}$. Hence, by Theorem 4.3, $M_{L,P} \leq M$. Therefore $M_{L,P}$ is a least $L$-model of $P$. □

[7] w.r.t. the pre-order between modalities specified in Definition 4.9

### 4.3 SLD-Resolution

The fixpoint semantics can be viewed as a bottom-up method for computing answers. It repeatedly applies clauses of a given program $P$ in order to compute the set $I_{L,P}$ of facts derivable in $L$ from the program. Given an atom $\alpha$ from $I_{L,P}$, the process of tracing back the derivation of $\alpha$ in $L$ from $P$ is called top-down, because it reduces the atom, treated as a goal, to subgoals. A more general problem is finding answers for an $L$-MProlog goal which are correct w.r.t. an $L$-MProlog program. This problem is reduced to the previous by unification and a lifting technique. An *SLD-resolution calculus* for $L$-MProlog will be developed to solve the problem.

The main work in developing an SLD-resolution calculus for $L$-MProlog is to specify a reverse of the operator $T_{L,P}$. While $T_{L,P}$ acts on model generators (with only ground atoms), the expected reverse of $T_{L,P}$ will act on goals (with variables). The operator $T_{L,P}$ is a composition of $Sat_L$, $T_{0L,P}$, and $NF_L$. So, we have to investigate reversion of these operators.

**Definition 4.18** A *goal* is a clause of the form $\leftarrow \alpha_1, \ldots, \alpha_k$, where $\alpha_i$ are atoms.

The following definition concerns reversion of the operator $T_{0L,P}$.

**Definition 4.19** Let $G = \leftarrow \alpha_1, \ldots, \alpha_i, \ldots, \alpha_k$ be a goal and $\phi = \triangle(A \leftarrow B_1, \ldots, B_l)$ a program clause. Then $G'$ is *derived* from $G$ and $\phi$ in $L$ using mgu $\theta$, and called an *L-resolvent* of $G$ and $\phi$, if the following conditions hold:

- $\alpha_i = \triangle' A'$, with $\triangle'$ in the $L$-normal labeled form, is called the *selected atom*, and $A'$ is called the *selected head atom*.
- $\triangle'$ is an $L$-instance of a universal modality $\triangle''$ and $\triangle''(A \leftarrow B_1, \ldots, B_l)$ is an $L$-instance of $\phi$.
- $\theta$ is an mgu of $A'$ and the forward labeled form of $A$.
- $G'$ is the goal $\leftarrow (\alpha_1, \ldots, \alpha_{i-1}, \triangle' B_1, \ldots, \triangle' B_l, \alpha_{i+1}, \ldots, \alpha_k)\theta$.

As a reverse of the operator $Sat_L$, we provide the operator $rSat_L$.

**Definition 4.20** The *operator $rSat_L$* is specified by a finite set of backward rules. We say that $\beta = rSat_L(\alpha)$ *using an $rSat_L$ rule* $\alpha' \leftarrow \beta'$ if $\alpha \leftarrow \beta$ is of the form $\alpha' \leftarrow \beta'$. We write $\beta = rSat_L(\alpha)$ to denote that $\beta = rSat_L(\alpha)$ using some $rSat_L$ rule.

We require that one of the $rSat_L$ rules is the *backward labeling rule* $\triangle \Diamond_i E \leftarrow \triangle \langle X \rangle_i E$ with $X$ being a fresh atom variable. We call $\triangle \langle X \rangle_i E$ a *backward labeled form* of $\triangle \Diamond_i E$.

**Definition 4.21** Let $G = \leftarrow \alpha_1, \ldots, \alpha_i, \ldots, \alpha_k$ be a goal. If $\alpha'_i = rSat_L(\alpha_i)$ using an $rSat_L$ rule $\phi$, then $G' = \leftarrow \alpha_1, \ldots, \alpha_{i-1}, \alpha'_i, \alpha_{i+1}, \ldots, \alpha_k$ is *derived* from $G$ and $\phi$, and we call $G'$ a *(L-)resolvent* of $G$ and $\phi$, and $\alpha_i$ the *selected atom* (of $G$).

As a reverse of the operator $NF_L$, we provide the operator $rNF_L$.

**Definition 4.22** The *operator $rNF_L$* is specified by a finite set of backward rules. We say that $\beta =_\theta rNF_L(\alpha)$ *using an $rNF_L$ rule* $\alpha' \leftarrow \beta'$ if $\theta$ is an mgu such that $\alpha\theta \leftarrow \beta$ is of the form $\alpha' \leftarrow \beta'$. We write $\beta =_\theta rNF_L(\alpha)$ if $\beta =_\theta rNF_L(\alpha)$ using some $rNF_L$ rule.

**Definition 4.23** Let $G = \leftarrow \alpha_1, \ldots, \alpha_i, \ldots, \alpha_k$ be a goal. Then $G'$ is *derived* from $G$ and an $rNF_L$ rule $\phi$ (in $L$) using mgu $\theta$, and called a *(L-)resolvent* of $G$ and $\phi$, if $\alpha_i$ is called the *selected atom*, $\alpha'_i =_\theta rNF_L(\alpha_i)$ using $\phi$, and $G' = \leftarrow \alpha_1\theta, \ldots, \alpha_{i-1}\theta, \alpha'_i, \alpha_{i+1}\theta, \ldots, \alpha_k\theta$.

Observe that $rSat_L$ rules and $rNF_L$ rules are similar to program clauses and the way of applying them is similar to the way of applying program clauses, except that for the logics considered in this work we do not need mgu's for $rSat_L$ rules. We now define SLD-derivation and SLD-refutation.

**Definition 4.24** Let $P$ be an $L$-MProlog program and $G$ a goal. An *SLD-derivation* of $P \cup \{G\}$ in $L$ consists of a (finite or infinite) sequence $G_0 = G, G_1, \ldots$ of goals, a sequence $\phi_1, \phi_2, \ldots$ of variants of program clauses of $P$, $rSat_L$ rules, or $rNF_L$ rules, and a sequence $\theta_1, \theta_2, \ldots$ of mgu's such that if $\phi_i$ is a variant of a program clause or an $rNF_L$ rule then $G_i$ is derived from $G_{i-1}$ and $\phi_i$ in $L$ using $\theta_i$, else $\theta_i = \epsilon$ and $G_i$ is derived from $G_{i-1}$ and (the $rSat_L$ rule variant) $\phi_i$.

We require that each $\phi_i$ in the above definition does not have any variable which already appears in the derivation up to $G_{i-1}$. This can be achieved by subscripting variables in $G$ by 0 and variables in $\phi_i$ by $i$. This process of renaming variables is usually called *standardizing* the variables *apart* (see [22]). Each $\phi_i$ is called an *input clause/rule* of the derivation.

**Definition 4.25** An *SLD-refutation* (*refutation* in short) of $P \cup \{G\}$ in $L$ is a finite SLD-derivation of $P \cup \{G\}$ in $L$ which has the empty clause $\diamond$ as the last goal in the derivation. If $G_n = \diamond$ then the refutation has *length* $n$.

**Definition 4.26** Let $P$ be an $L$-MProlog program and $G$ a goal. A *computed answer* $\theta$ in $L$ for $P \cup \{G\}$ is the substitution obtained by restricting the composition $\theta_1 \ldots \theta_n$ to the variables of $G$, where $\theta_1, \ldots, \theta_n$ is the sequence of mgu's used in an SLD-refutation of $P \cup \{G\}$ in $L$.

## 4.4 Soundness and Completeness of SLD-Resolution

This subsection concerns soundness and completeness of SLD-resolution. Informally, an SLD-resolution calculus is sound if every computed answer for $P \cup \{G\}$ is a correct answer for $P \cup \{G\}$, and is complete if for every correct answer for $P \cup \{G\}$ there exists a computed answer for $P \cup \{G\}$ that is more general.

**Definition 4.27** We say that an atom $\beta$ is *derivable* from $\alpha$ using $rSat_L$ (resp. (i) $rNF_L$, (ii) $rSat_L$ and $rNF_L$) if there exists a sequence of atoms $\alpha_0, \ldots, \alpha_k$ with $k \geq 0$, $\alpha_0 = \alpha$ and $\alpha_k = \beta$ such that for every $1 \leq i \leq k$, $\alpha_i = rSat_L(\alpha_{i-1})$ (resp. (i) $\alpha_i =_{\theta_i} rNF_L(\alpha_{i-1})$ for some $\theta_i$, (ii) $\alpha_i = rSat_L(\alpha_{i-1})$ or $\alpha_i =_{\theta_i} rNF_L(\alpha_{i-1})$ for some $\theta_i$).

The main results are proved using the following expected properties of $rSat_L$ and $rNF_L$:

**Lemma 4.10 (to be proved)** *Let $\triangle$ and $\triangle'$ be ground modalities in the L-normal labeled form. Let $B$ be an atom of the form $E$, $\Diamond_i E$, or $\Box_i E$, and $B'$ an atom of the form $E$, $\Diamond_j E$, $\langle X \rangle_j E$, or $\Box_j E$. Suppose that $\triangle$ is an L-instance of $\triangle'$ and $B$ is an L-instance of $B'$. Then $\triangle' B'$ is derivable from $\triangle B$ using $rSat_L$.*

**Lemma 4.11 (to be proved)** *Let $\alpha \in Sat_L(\{\beta\})$ or $\alpha \in NF_L(\{\beta\})$. Then there exists an atom $\beta'$ and a substitution $\theta$ such that $\beta = \beta'\theta$, the domain of $\theta$ consists of fresh atom variables, and $\beta'$ is derivable from $\alpha$ using $rSat_L$ and $rNF_L$.*

**Lemma 4.12 (to be proved)** *Let $\beta = rSat_L(\alpha)$, $M$ be an L-model, $\sigma$ a $\Diamond$-realization function on $M$, and $\theta$ a substitution. Suppose that $M, \sigma \vDash \forall_c(\beta'\theta)$ for some $\Box$-lifting form $\beta'$ of $\beta$. Then $M, \sigma \vDash \forall_c(\alpha'\theta)$ for some $\Box$-lifting form $\alpha'$ of $\alpha$.*

**Lemma 4.13 (to be proved)** *Let $\beta =_\delta rNF_L(\alpha)$, $M$ be an L-model, $\sigma$ a maximal $\Diamond$-realization function on $M$, and $\theta$ a substitution. Suppose that $M, \sigma \vDash \forall_c(\beta'\theta)$ for some $\Box$-lifting form $\beta'$ of $\beta$. Then $M, \sigma \vDash \forall_c(\alpha'\delta\theta)$ for some $\Box$-lifting form $\alpha'$ of $\alpha$.*

### 4.4.1 Soundness

We first prove the following auxiliary lemma.

**Lemma 4.14** *Let $M$ be a Kripke model, $\sigma$ a $\Diamond$-realization function on $M$, and $\theta$ a substitution. Suppose that $\triangle^{(1)}, \ldots, \triangle^{(l)}$ are $\Box$-lifting forms of $\triangle$ and $M, \sigma \vDash \forall_c((\triangle^{(1)} B_1 \wedge \ldots \wedge \triangle^{(l)} B_l)\theta)$. Then there exists the most general L-instance $\triangle'$ of $\triangle^{(1)}, \ldots, \triangle^{(l)}$, which is a $\Box$-lifting form of $\triangle$ and satisfies $M, \sigma \vDash \forall_c((\triangle' B_1 \wedge \ldots \wedge \triangle' B_l)\theta)$.*

*Proof.* Let $h = |\triangle|$. For $1 \leq j \leq l$ and $1 \leq k \leq h$, let $\nabla^{(j,k)}$ be the modal operator at position $k$ of $\triangle^{(j)}$, and $\nabla^{(k)}$ the modal operator at position $k$ of $\triangle$. Let $i_k$ be the number indicating the kind of the modal operator $\nabla^{(k)}$. If $\nabla^{(j,k)} = \Box_{i_k}$ for all $1 \leq j \leq l$, then let $\nabla^{(k')} = \Box_{i_k}$, else let $\nabla^{(k')} = \nabla^{(k)}$. Let $\triangle' = \nabla^{(1')} \ldots \nabla^{(h')}$. Clearly, $\triangle'$ is the most general $L$-instance of $\triangle^{(1)}, \ldots, \triangle^{(l)}$ and is a $\Box$-lifting form of $\triangle$.

Because that for $1 \leq j \leq l$, $\triangle^{(j)}$ is a $\Box$-lifting form of $\triangle'$ and $M, \sigma \vDash \forall_c((\triangle^{(j)} B_j)\theta)$, it can be proved by induction on $k$ that $M, \sigma \vDash \forall_c((\nabla^{(1')} \ldots \nabla^{(k')} \top)\theta)$, for $1 \leq k \leq h$. It follows that $M, \sigma \vDash \forall_c((\triangle' \top)\theta)$. Because $\triangle^{(j)}$ is a $\Box$-lifting form of $\triangle'$, for $1 \leq j \leq l$, and $M, \sigma \vDash \forall_c((\triangle^{(1)} B_1 \wedge \ldots \wedge \triangle^{(l)} B_l)\theta)$, we conclude that $M, \sigma \vDash \forall_c((\triangle' B_1 \wedge \ldots \wedge \triangle' B_l)\theta)$. $\Box$

The soundness theorem is based on the following lemma:

**Lemma 4.15** *Let $P$ be an $L$-MProlog program and $G = \leftarrow \alpha_1, \ldots, \alpha_k$ a goal. Then for every computed answer $\theta$ in $L$ for $P \cup \{G\}$ there exists a goal $G' = \leftarrow \alpha'_1, \ldots, \alpha'_k$ such that $\alpha'_i$ is a $\Box$-lifting form of $\alpha_i$, for $1 \leq i \leq k$, and $P \vDash_L \forall_c((\alpha'_1 \wedge \ldots \wedge \alpha'_k)\theta)$.*

*Proof.* Let $M$ be an arbitrary $L$-model of $P$ and $\sigma$ a maximal $\Diamond$-realization function on $M$. Let the refutation of $P \cup \{G\}$ in $L$ consist of a sequence $G_0 = G, G_1, \ldots, G_n$ of goals, a sequence $\phi_1, \ldots, \phi_n$ of variants of program clauses of $P$, $rSat_L$ rules, or $rNF_L$ rules, and a sequence $\theta_1, \ldots, \theta_n$ of mgu's. Let $\theta$ be the computed answer. We prove by induction on $n$ that for every $1 \leq i \leq k$ there exists a $\Box$-lifting form $\alpha'_i$ of $\alpha_i$ such that $M, \sigma \vDash \forall_c((\alpha'_1 \wedge \ldots \wedge \alpha'_k)\theta)$.

Suppose that $n = 1$. This means that $G = \leftarrow \alpha_1$ with $\alpha_1 = \triangle' A'$, $A'$ is the selected head atom, and the empty clause is an $L$-resolvent of $G$ and some input clause $\phi_1 = \triangle(A \leftarrow)$. By Lemma 4.4, $P \vDash_L \forall(\Box_{i_1} \ldots \Box_{i_h} A)$, where $\Box_{i_1} \ldots \Box_{i_h}$ is a $\Box$-lifting form of $\triangle'$. If $A'$ is of the form $\Box_i E$ or $E$, then $A'\theta_1 = A\theta_1$, and $P \vDash_L \forall(\Box_{i_1} \ldots \Box_{i_h} A'\theta_1)$. Suppose that $A' = \langle F \rangle_i E'$ or $A' = \langle X \rangle_i E'$. Thus $A$ is of the form $\Diamond_i E$. Let $A'' = \langle E \rangle_i E$ (the forward labeled form of $A$). We have $A'\theta_1 = A''\theta_1 = \langle E'' \rangle_i E''$ for some $E''$. Since $P \vDash_L \forall(\Box_{i_1} \ldots \Box_{i_h} A)$, we have $P \vDash_L \forall(\Box_{i_1} \ldots \Box_{i_h} \Diamond_i E'')$. It follows that $M, \sigma \vDash \forall_c(\Box_{i_1} \ldots \Box_{i_h} \langle E'' \rangle_i E'')$, hence $M, \sigma \vDash \forall_c(\Box_{i_1} \ldots \Box_{i_h} A'\theta_1)$. Hence, for $\alpha'_1 = \Box_{i_1} \ldots \Box_{i_h} A'$, we have $M, \sigma \vDash \forall_c(\alpha'_1 \theta)$.

Next suppose that the result holds for computed answers which come from refutations of length less than $n$. There are the following cases: $G_1$ is derived from $G$ and an $rSat_L/rNF_L$ rule variant, or $G_1$ is an $L$-resolvent of $G$ and a variant of some program clause of $P$. The case $G_1$ is derived from $G$ and an $rSat_L$ rule variant immediately follows from the inductive assumption and Lemma 4.12.

Suppose that $G_1$ is derived from $G$ and an $rNF_L$ rule variant, $\alpha_i$ is the selected atom and it is replaced by $\beta =_{\theta_1} rNF_L(\alpha_i)$. We have

$$G_1 = \leftarrow \alpha_1\theta_1, \ldots, \alpha_{i-1}\theta_1, \beta, \alpha_{i+1}\theta_1 \ldots, \alpha_k\theta_1$$

By the inductive assumption, there exist a $\Box$-lifting form $\alpha'_j$ of $\alpha_j$, for $1 \leq j \leq k$ and $j \neq i$, and a $\Box$-lifting form $\beta'$ of $\beta$ such that

$$M, \sigma \vDash \forall_c((\alpha'_1\theta_1 \wedge \ldots \wedge \alpha'_{i-1}\theta_1 \wedge \beta' \wedge \alpha'_{i+1}\theta_1 \wedge \ldots \wedge \alpha'_k\theta_1)\theta_2 \ldots \theta_n)$$

We have $M, \sigma \vDash \forall_c(\beta'\theta_2 \ldots \theta_n)$. Hence, by Lemma 4.13, there exists a $\Box$-lifting form $\alpha'_i$ of $\alpha_i$ such that $M, \sigma \vDash \forall_c(\alpha'_i\theta_1\theta_2 \ldots \theta_n)$. Therefore $M, \sigma \vDash \forall_c((\alpha'_1 \wedge \ldots \wedge \alpha'_k)\theta)$.

Now suppose that $G_1$ is derived in $L$ from $G$ and an input clause $\phi = \triangle(A \leftarrow B_1, \ldots, B_l)$ ($l \geq 0$), the selected atom is $\alpha_i = \triangle' A'$, and $A'$ is the selected head atom. We have

$$G_1 = \leftarrow (\alpha_1, \ldots, \alpha_{i-1}, \triangle' B_1, \ldots, \triangle' B_l, \alpha_{i+1}, \ldots, \alpha_k)\theta_1$$

By the inductive assumption, there exists a goal

$$G'_1 = \leftarrow (\alpha'_1, \ldots, \alpha'_{i-1}, \triangle^{(1')} B'_1, \ldots, \triangle^{(l')} B'_l, \alpha'_{i+1}, \ldots, \alpha'_k)\theta_1$$

such that

$$M, \sigma \vDash \forall_c((\alpha'_1 \wedge \ldots \wedge \alpha'_{i-1} \wedge \triangle^{(1')} B'_1 \wedge \ldots \wedge \triangle^{(l')} B'_l \wedge \alpha'_{i+1} \wedge \ldots \wedge \alpha'_k)\theta)$$

where $\alpha'_j$ is a $\Box$-lifting form of $\alpha_j$, for $1 \leq j \leq k$ and $j \neq i$, and $\triangle^{(j')}B'_j$ is a $\Box$-lifting form of $\triangle' B_j$ with $|\triangle^{(j')}| = |\triangle'|$, for $1 \leq j \leq l$. Let $\Box_{i_1} \ldots \Box_{i_h}$ be a $\Box$-lifting form of $\triangle'$, and $\triangle''$ be the most general $L$-instance of $\triangle^{(1')}, \ldots, \triangle^{(l')}$ if $l > 0$, which exists due to Lemma 4.14, and be $\Box_{i_1} \ldots \Box_{i_h}$ otherwise. By Lemma 4.14, $\triangle''$ is a $\Box$-lifting form of $\triangle'$, and $M, \sigma \models \forall_c((\triangle'' B'_1 \wedge \ldots \wedge \triangle'' B'_l)\theta)$ if $l > 0$. Since $M$ is an $L$-model of $P$, by Lemma 4.4, we have $M \models \forall(\Box_{i_1} \ldots \Box_{i_h}(B_1 \wedge \ldots \wedge B_l \rightarrow A))$. Hence $M, \sigma \models \forall_c((\triangle'' A)\theta)$ (because $\Box_{i_1} \ldots \Box_{i_h}$ is a $\Box$-lifting form of $\triangle''$, $B'_j$ is a $\Box$-lifting form of $B_j$, and $L$ is a serial modal logic). Let $A''$ be the forward labeled form of $A$. It follows that $M, \sigma \models \forall_c((\triangle'' A'')\theta)$. Since $A'\theta_1 = A''\theta_1$, by choosing $\alpha'_i = \triangle'' A'$, we have that $\alpha'_i$ is a $\Box$-lifting form of $\alpha_i$ and $M, \sigma \models \forall_c((\alpha'_1 \wedge \ldots \wedge \alpha'_k)\theta)$. This completes the proof. $\Box$

**Theorem 4.16 (Soundness of SLD-Resolution)** *Let $P$ be an $L$-MProlog program and $G$ an $L$-MProlog goal. Then every computed answer in $L$ for $P \cup \{G\}$ is a correct answer in $L$ for $P \cup \{G\}$.*

*Proof.* Let $G = \leftarrow \alpha_1, \ldots, \alpha_k$, where each $\alpha_i$ is of the form $\triangle E$ or $\triangle \Diamond E$, with $\triangle$ being a universal modality. Let $\theta$ be a computed answer in $L$ for $P \cup \{G\}$. Since $L$ is a serial modal logic, by Lemma 4.15, we have $P \models_L \forall_c((\alpha_1 \wedge \ldots \wedge \alpha_k)\theta)$. Assume that the signature contains enough constant symbols, for example, infinitely many. Then it follows that $P \models_L \forall((\alpha_1 \wedge \ldots \wedge \alpha_k)\theta)$. Hence $\theta$ is a correct answer in $L$ for $P \cup \{G\}$. $\Box$

### 4.4.2 Completeness

We use a standard method to prove completeness of our SLD-resolution calculus (cf. [22, 21]). In general, completeness of a resolution calculus is first proved for the ground version and then lifted to the case with variables. The flow of this section follows Lloyd [22]. The proofs of Lemmas 4.17, 4.18, 4.22 and Theorem 4.23 are very similar to the ones given for classical logic programming in Lloyd's book and are presented in Appendix A.

We first define unrestricted SLD-refutation and give the *mgu lemma* and the *lifting lemma*, which are proved in Appendix A.

**Definition 4.28** An *unrestricted SLD-refutation* in $L$ is an SLD-refutation in $L$, except that we drop the requirement that the substitutions $\theta_i$ be most general unifiers. They are only required to be unifiers. In an unrestricted SLD-resolution, if a goal $G_i$ is derived from $G_{i-1}$ and an $rSat_L$ rule variant, then $\theta_i$ can be arbitrary and $G_i = G'_i\theta_i$, where $G'_i$ is the goal derived from $G_{i-1}$ and that $rSat_L$ rule variant in the usual way.

**Lemma 4.17 (Mgu Lemma)** *Let $P$ be an $L$-MProlog program and $G$ a goal. Suppose that $P \cup \{G\}$ has an unrestricted SLD-refutation in $L$. Then $P \cup \{G\}$ has an SLD-refutation in $L$ of the same length such that, if $\theta_1, \ldots, \theta_n$ are the unifiers from the unrestricted refutation and $\theta'_1, \ldots, \theta'_n$ are mgu's from the refutation, then there exists a substitution $\gamma$ such that $\theta_1 \ldots \theta_n = \theta'_1 \ldots \theta'_n \gamma$.*

**Lemma 4.18 (Lifting Lemma)** *Let $P$ be an $L$-MProlog program, $G$ a goal, and $\theta$ a substitution. Suppose there exists an SLD-refutation of $P \cup \{G\theta\}$ in $L$. Then there exists an SLD-refutation of $P \cup \{G\}$ in $L$ of the same length such that, if $\theta_1, \ldots, \theta_n$ are the mgu's from the refutation of $P \cup \{G\theta\}$ and $\theta'_1, \ldots, \theta'_n$ are the mgu's from the refutation of $P \cup \{G\}$, then there exists a substitution $\gamma$ such that $\theta\theta_1 \ldots \theta_n = \theta'_1 \ldots \theta'_n \gamma$.*

The following lemma is an essential part of the completeness proof.

**Lemma 4.19** *Let $P$ be an $L$-MProlog program and $\alpha \in I_{L,P}$. Then $P \cup \{\leftarrow \alpha\}$ has an SLD-refutation in $L$.*

*Proof.* We prove by induction on $n$ that if $\alpha \in T_{L,P} \uparrow n$, then $P \cup \{\leftarrow \alpha\}$ has an SLD-refutation in $L$. This assertion obviously holds for $n = 0$ since $T_{L,P} \uparrow 0 = \emptyset$.

Suppose that the assertion holds for $(n-1)$ in the place of $n$. Let $\alpha \in T_{L,P} \uparrow n$. There exist a variant $\phi = \triangle(A \leftarrow B_1, \ldots, B_k)$ of a program clause of $P$, with $k \geq 0$, a substitution $\theta$, modalities $\triangle'$ and $\triangle''$, ground atoms $\gamma_1, \ldots, \gamma_k \in T_{L,P} \uparrow (n-1)$, and ground atoms $\beta_1, \ldots, \beta_k$, $\alpha'$ such that:

- $\beta_i \in Sat_L(\{\gamma_i\})$, for $1 \le i \le k$;
- $\beta_i = \triangle^{(i)} B'_i$ and $B_i\theta$ is an $L$-instance of $B'_i$, for $1 \le i \le k$;
- $\triangle''(A \leftarrow B_1, \ldots, B_k)$ is an $L$-instance of the input clause $\phi$;
- $\triangle'$ is in the $L$-normal labeled form and is an $L$-instance of $\triangle^{(1)}, \ldots, \triangle^{(k)}, \triangle''$;
- $\alpha' = \triangle' A'\theta$, where $A'$ is the forward labeled form of $A$;
- $\alpha \in NF_L(\{\alpha'\})$.

By Lemma 4.11, there exist atoms $\alpha''$, $\gamma'_1, \gamma'_2, \ldots, \gamma'_k$, and ground substitutions $\delta_0, \ldots, \delta_k$ with disjoint domains such that:

- $\alpha''$ is derivable from $\alpha$ using $rSat_L$ and $rNF_L$, and $\alpha' = \alpha''\delta_0$,
- $\gamma'_i$ is derivable from $\beta_i$ using $rSat_L$ and $rNF_L$, and $\gamma_i = \gamma'_i\delta_i$, for $1 \le i \le k$.

Let $\delta = \delta_1 \ldots \delta_k$ if $k > 0$, and $\delta = \varepsilon$ otherwise. By the inductive assumption, $P \cup \{\leftarrow \gamma_i\}$ has a refutation in $L$, for $1 \le i \le k$. Since $\gamma'_i\delta = \gamma'_i\delta_i = \gamma_i$, it follows that $P \cup \{\leftarrow \gamma'_i\delta\}$ has a refutation in $L$. Hence $P \cup \{\leftarrow (\gamma'_1, \ldots, \gamma'_k)\delta\}$ has a refutation in $L$, since $\gamma'_i\delta$ are ground. By the lifting lemma, $P \cup \{\leftarrow \gamma'_1, \ldots, \gamma'_k\}$ has a refutation in $L$. Since $\gamma'_i$ is derivable from $\beta_i$ using $rSat_L$ and $rNF_L$, it follows that $P \cup \{\leftarrow \beta_1, \ldots, \beta_k\}$ has a refutation in $L$.

For $1 \le i \le k$, if $B'_i$ is of the form $\langle F \rangle_h E$ then let $\beta'_i = \triangle^{(i)}\langle X \rangle_h E$ and $\theta_i = \{X/F\}$, where $X$ is a fresh atom variable; else let $\beta'_i = \beta_i$ and $\theta_i = \varepsilon$. Let $\theta' = \theta_1 \ldots \theta_k$ if $k > 0$, and $\theta' = \varepsilon$ otherwise. Since $\beta_i = \beta'_i\theta'$, $P \cup \{\leftarrow (\beta'_1, \ldots, \beta'_k)\theta'\}$ has a refutation in $L$. Hence, by the lifting lemma, $P \cup \{\leftarrow \beta'_1, \ldots, \beta'_k\}$ has a refutation in $L$. Therefore, by Lemma 4.10, $P \cup \{\leftarrow \triangle' B_1\theta, \ldots, \triangle' B_k\theta\}$ has a refutation in $L$. The goal $\leftarrow \triangle' B_1\theta, \ldots, \triangle' B_k\theta$ is an unrestricted $L$-resolvent of $\leftarrow \alpha'$ and $\phi$. Hence, by the mgu lemma, $P \cup \{\leftarrow \alpha'\}$ has a refutation in $L$. This means that $P \cup \{\leftarrow \alpha''\delta_0\}$ has a refutation in $L$. By the lifting lemma, $P \cup \{\leftarrow \alpha''\}$ has a refutation in $L$. Since $\alpha''$ is derivable from $\alpha$ using $rSat_L$ and $rNF_L$, we conclude that $P \cup \{\leftarrow \alpha\}$ has a refutation in $L$. □

**Corollary 4.20** *Let $P$ be an $L$-MProlog program and $\alpha \in Sat_L(I_{L,P})$. Then $P \cup \{\leftarrow \alpha\}$ has an SLD-refutation in $L$.*

*Proof.* There exists $\beta \in I_{L,P}$ such that $\alpha \in Sat_L(\{\beta\})$. By Lemma 4.11, there exist an atom $\beta'$ and a substitution $\theta$ such that $\beta = \beta'\theta$ and $\beta'$ is derivable from $\alpha$ using $rSat_L$ and $rNF_L$. Since $\beta \in I_{L,P}$, by Lemma 4.19, $P \cup \{\leftarrow \beta\}$ has a refutation in $L$. This means that $P \cup \{\leftarrow \beta'\theta\}$ has a refutation in $L$. By the lifting lemma, $P \cup \{\leftarrow \beta'\}$ has a refutation in $L$. Consequently, $P \cup \{\leftarrow \alpha\}$ has a refutation in $L$. □

**Lemma 4.21** *Let $P$ be an $L$-MProlog program and $\alpha$ a ground $L$-MProlog goal atom such that $M_{L,P} \models \alpha$. Then $P \cup \{\leftarrow \alpha\}$ has an SLD-refutation in $L$.*

*Proof.* By Lemma 4.5, $\alpha$ is an $L$-instance of some $\alpha' \in Sat_L(I_{L,P})$. By Corollary 4.20, $P \cup \{\leftarrow \alpha'\}$ has an SLD-refutation in $L$. If $\alpha'$ is of the form $E$, $\triangle\Diamond_i E$, or $\triangle\Box_i E$ then, by Lemma 4.10, $P \cup \{\leftarrow \alpha\}$ has an SLD-refutation in $L$. If $\alpha'$ is of the form $\triangle\langle F \rangle_i E$ then, by the lifting lemma, $P \cup \{\leftarrow \triangle\langle X \rangle_i E\}$ has an SLD-refutation in $L$, where $X$ is a fresh atom variable. By the assumption about $\preceq_L$, $\alpha$ is also an $L$-instance of $\triangle\langle X \rangle_i E$. Hence, by Lemma 4.10, $P \cup \{\leftarrow \alpha\}$ has an SLD-refutation in $L$. □

For the main theorem, we need another auxiliary lemma, whose proof uses Lemma 4.21 and is presented in Appendix A.

**Lemma 4.22** *Let $P$ be an $L$-MProlog program and $\alpha$ an $L$-MProlog goal atom. Suppose that $\forall(\alpha)$ is a logical consequence in $L$ of $P$. Then there exists an SLD-refutation of $P \cup \{\leftarrow \alpha\}$ in $L$ with the identity substitution as the computed answer.*

The following completeness theorem follows from the above lemma and the lifting lemma. Its proof is given in Appendix A.

**Theorem 4.23 (Completeness of SLD-Resolution)** *Let $P$ be an $L$-MProlog program and $G$ an $L$-MProlog goal. For every correct answer $\theta$ in $L$ for $P \cup \{G\}$, there exists a computed answer $\gamma$ in $L$ for $P \cup \{G\}$ and a substitution $\delta$ such that $\theta = \gamma\delta$.*

| $L = BSMM$, $L$-MProlog |
|---|
| $\preceq_L$ is defined by Definition 4.8.<br><br>No restrictions on the $L$-normal form of modalities.<br>No rules specifying $NF_L$ and $rNF_L$.<br><br>Rules specifying<br>$Ext_L$ and $Sat_L$<br>$\triangle\langle E\rangle_i\alpha \to \triangle\Diamond_i\alpha$<br>$\triangle\Box_i\alpha \to \triangle\Diamond_i\alpha$<br>$\triangle\Box_i\alpha \to \triangle\alpha$ if $AT(i)$<br>$\triangle\alpha \to \triangle\Diamond_i\alpha$ if $AT(i)$<br>$\triangle\Box_i\alpha \to \triangle\Box_j\alpha$ if $AI(i,j)$<br>$\triangle\Diamond_j\alpha \to \triangle\Diamond_i\alpha$ if $AI(i,j)$<br>$\triangle\alpha \to \triangle\Box_i\Diamond_j\alpha$ if $AB(i,j)$<br>$\triangle\Diamond_i\Box_j\alpha \to \triangle\alpha$ if $AB(i,j)$<br>$\triangle\Box_i\alpha \to \triangle\Box_j\Box_k\alpha$ if $A4(i,j,k)$<br>$\triangle\Diamond_j\Diamond_k\alpha \to \triangle\Diamond_i\alpha$ if $A4(i,j,k)$<br>$\triangle\Diamond_i\alpha \to \triangle\Box_j\Diamond_k\alpha$ if $A5(i,j,k)$<br>$\triangle\Diamond_j\Box_k\alpha \to \triangle\Box_i\alpha$ if $A5(i,j,k)$<br>$rSat_L$<br>$\triangle\Diamond_i\alpha \leftarrow \triangle\langle X\rangle_i\alpha$ where $X$ is a fresh atom variable<br>$\triangle\nabla_i\alpha \leftarrow \triangle\Box_i\alpha$<br>plus a rule $\alpha \leftarrow \beta$ for each $k$-th rule $\beta \to \alpha$ specifying $Sat_L$,<br>$k \geq 3$, with the same accompanying condition |

Table 1: A schema for semantics of $BSMM$-MProlog

### 4.5 Summary

We have given a general framework for developing fixpoint semantics, the least model semantics, and SLD-resolution calculi for $L$-MProlog programs. The base logic $L$ is required so that the $L$-frame restrictions consist of $\forall x\ \exists y\ R_i(x,y)$ (seriality), for $1 \leq i \leq m$, and some classical first-order Horn clauses.

**Definition 4.29** By a *schema for semantics of L-MProlog* we mean a table of definitions of the $L$-normal form of modalities and $\preceq_L$ plus rules specifying the operators $Ext_L$, $Sat_L$, $NF_L$, $rNF_L$, $rSat_L$. We say that such a schema is *correct* if all the results of this section hold for $L$-MProlog w.r.t. that schema.

To show correctness of a schema, we have to prove the lemmas and theorems marked as "to be proved", i.e. Theorem 4.3 and Lemmas 4.2, 4.4, 4.5, 4.7, 4.8, 4.10 – 4.13.

## 5 A Schema for Semantics of $BSMM$-MProlog

In this section, let $L$ be a $BSMM$ logic. In Table 1, we give a schema for semantics of $BSMM$-MProlog. The first rule specifying $rSat_L$ is a generalized version of the backward labeled rule and it is dual to the first rule specifying $Sat_L$. The remaining rules specifying $Sat_L$ and $rSat_L$ directly come from the axioms. The schema given in Table 1 suggests a general method for translating axioms of a given modal logic into an SLD-resolution calculus for that logic.

**Theorem 5.1** *The schema for semantics of BSMM-MProlog is correct.*

We now prove this theorem. Lemmas 4.4 and 4.10 – 4.13 can be easily verified. To prove the remaining lemmas and theorems we need *extended L-model graphs* (defined below) and some properties of them.

**Definition 5.1** Let $I$ be an $L$-model generator. Let $Ext'_L$ be the operator such that $Ext'_L(I)$ is the least set of atoms containing $I$ and closed w.r.t. the rules specifying $Ext_L$. The *extended L-model graph* of $I$ is defined in the same way as the standard $L$-model graph of $I$ but with $Ext'_L(I)$ in the place of $Ext_L(I)$.

We have $Ext_L(I) \subseteq Ext'_L(I)$.

**Lemma 5.2** *Let $I$ be an $L$-model generator, $M$ the standard $L$-model graph of $I$, and $M'$ the extended $L$-model graph of $I$. Then $M'$ has the same frame as $M$, and furthermore, if $M = \langle W, \tau, R_1, \ldots, R_m, H\rangle$ and $M' = \langle W, \tau, R_1, \ldots, R_m, H'\rangle$ then for any $w \in W$, $H(w) \subseteq H'(w)$ and $H'(w) - H(w)$ is a set of formulas containing some unlabeled existential modal operators.*

The proof of this lemma is straightforward.

The following lemma is similar to Lemma 4.1 and its proof is also straightforward.

**Lemma 5.3** *Let $I$ be an $L$-normal model generator and $M = \langle W, \tau, R_1, \ldots, R_m, H\rangle$ the extended $L$-model graph of $I$. Let $w = \langle E_1\rangle_{i_1} \ldots \langle E_k\rangle_{i_k}$ be a world of $M$ and $\triangle = w$ be a modality. Then for $\alpha$ (resp. $A$) not containing $\top$, $\alpha \in H(w)$ (resp. $A \in H(w)$) iff there exists a $\Box$-lifting form $\triangle'$ of $\triangle$ such that $\triangle'\alpha \in Ext'_L(I)$ (resp. $\triangle' A \in Sat_L(I)$).*

**Lemma 5.4** *Let $I$ be an $L$-normal model generator and $M = \langle W, \tau, R_1, \ldots, R_m, H\rangle$ the extended $L$-model graph of $I$. Then for any $w$ and $u$ such that $R_i(w,u)$: i) if $\Box_i\alpha \in H(w)$ then $\alpha \in H(u)$, ii) if $\alpha \in H(u)$ then $\Diamond_i\alpha \in H(w)$.*

*Proof.* Let $\{R'_j \mid 1 \leq j \leq m\}$ be the skeleton of $M$. We prove this lemma by induction on the number of steps needed to obtain $R_i(w,u)$ when extending $\{R'_j \mid 1 \leq j \leq m\}$ to $\{R_j \mid 1 \leq j \leq m\}$.

Consider the first assertion. Suppose that $\Box_i\alpha \in H(w)$. By Lemma 5.3, there exists a $\Box$-lifting form $\triangle$ of $w$ such that $\triangle\Box_i\alpha \in Ext'_L(I)$. Since $R_i(w,u)$ holds, there are the following cases to consider:

- Case $u = w\langle E\rangle_i$ and $R'_i(w, w\langle E\rangle_i)$ : The assertion holds by the definition of $M$.
- Case $AT(i)$ holds and $u = w$ : Since $\triangle\Box_i\alpha \in Ext'_L(I)$, we have $\triangle\alpha \in Ext'_L(I)$, and by Lemma 5.3, $\alpha \in H(u)$.
- Case $AI(i,j)$ holds and $R_i(w,u)$ is created from $R_j(w,u)$ : Since $\triangle\Box_i\alpha \in Ext'_L(I)$, we have $\triangle\Box_j\alpha \in Ext'_L(I)$, and by Lemma 5.3, $\Box_j\alpha \in H(w)$. Hence, by the inductive assumption, $\alpha \in H(u)$.
- Case $AB(j,i)$ holds and $R_i(w,u)$ is created from $R_j(u,w)$ : Since $\Box_i\alpha \in H(w)$, by the inductive assumption, $\Diamond_j\Box_i\alpha \in H(u)$. By Lemma 5.3, there exists a $\Box$-lifting form $\triangle'$ of $u$ such that $\triangle'\Diamond_j\Box_i\alpha \in Ext'_L(I)$. Thus $\triangle'\alpha \in Ext'_L(I)$. Hence, by Lemma 5.3, $\alpha \in H(u)$.
- Case $A4(i,j,k)$ holds and $R_i(w,u)$ is created from $R_j(w,v)$ and $R_k(v,u)$: Since $\triangle\Box_i\alpha \in Ext'_L(I)$, we have $\triangle\Box_j\Box_k\alpha \in Ext'_L(I)$, and by Lemma 5.3, $\Box_j\Box_k\alpha \in H(w)$. Hence, by the inductive assumption, $\Box_k\alpha \in H(v)$ and $\alpha \in H(u)$.
- Case $A5(j,k,i)$ holds and $R_i(w,u)$ is created from $R_j(v,u)$ and $R_k(v,w)$: Since $\Box_i\alpha \in H(w)$, by the inductive assumption, $\Diamond_k\Box_i\alpha \in H(v)$. Hence, by Lemma 5.3, there exists a $\Box$-lifting form $\triangle'$ of $v$ such that $\triangle'\Diamond_k\Box_i\alpha \in Ext'_L(I)$. Hence $\triangle'\Box_j\alpha \in Ext'_L(I)$, and by Lemma 5.3, $\Box_j\alpha \in H(v)$. By the inductive assumption, it follows that $\alpha \in H(u)$.

Consider the second assertion. Suppose that $\alpha \in H(u)$. By Lemma 5.3, there exists a $\Box$-lifting form $\triangle$ of $u$ such that $\triangle\alpha \in Ext'_L(I)$. Since $R_i(w,u)$ holds, there are the following cases to consider:

- Case $u = w\langle E\rangle_i$ and $R'_i(w, w\langle E\rangle_i)$ : Since $\alpha \in H(u)$, either $\Box_i\alpha$ or $\langle E\rangle_i\alpha$ belongs to $H(w)$. Denote the formula by $\beta$. By Lemma 5.3, there exists a $\Box$-lifting form $\triangle'$ of $w$ such that $\triangle'\beta \in Ext'_L(I)$. Hence $\triangle'\Diamond_i\alpha \in Ext'_L(I)$, and by Lemma 5.3, $\Diamond_i\alpha \in H(w)$.

- Case $AT(i)$ holds and $u = w$ : Since $\triangle\alpha \in Ext'_L(I)$, $\triangle\Diamond_i\alpha \in Ext'_L(I)$, and by Lemma 5.3, $\Diamond_i\alpha \in H(w)$.

- Case $AI(i, j)$ holds and $R_i(w, u)$ is created from $R_j(w, u)$ : Since $\alpha \in H(u)$, by the inductive assumption, $\Diamond_j\alpha \in H(w)$. By Lemma 5.3, there exists a $\Box$-lifting form $\triangle'$ of $w$ such that $\triangle'\Diamond_j\alpha \in Ext'_L(I)$. Thus $\triangle'\Diamond_i\alpha \in Ext'_L(I)$. Hence, by Lemma 5.3, $\Diamond_i\alpha \in H(w)$.

- Case $AB(j, i)$ holds and $R_i(w, u)$ is created from $R_j(u, w)$ : Since $\triangle\alpha \in Ext'_L(I)$, we have $\triangle\Box_j\Diamond_i\alpha \in Ext'_L(I)$. Hence, by Lemma 5.3, $\Box_j\Diamond_i\alpha \in H(u)$. By the inductive assumption, it follows that $\Diamond_i\alpha \in H(w)$.

- Case $A4(i, j, k)$ holds and $R_i(w, u)$ is created from $R_j(w, v)$ and $R_k(v, u)$: Since $\alpha \in H(u)$, by the inductive assumption, $\Diamond_k\alpha \in H(v)$ and $\Diamond_j\Diamond_k\alpha \in H(w)$. By Lemma 5.3, there exists a $\Box$-lifting form $\triangle'$ of $w$ such that $\triangle'\Diamond_j\Diamond_k\alpha \in Ext'_L(I)$. Thus $\triangle'\Diamond_i\alpha \in Ext'_L(I)$, and by the Lemma 5.3, $\Diamond_i\alpha \in H(w)$.

- Case $A5(j, k, i)$ holds and $R_i(w, u)$ is created from $R_j(v, u)$ and $R_k(v, w)$: Since $\alpha \in H(u)$, by the inductive assumption, $\Diamond_j\alpha \in H(v)$. By Lemma 5.3, there exists a $\Box$-lifting form $\triangle'$ of $v$ such that $\triangle'\Diamond_j\alpha \in Ext'_L(I)$. Thus $\triangle'\Box_k\Diamond_i\alpha \in Ext'_L(I)$, and by the Lemma 5.3, $\Box_k\Diamond_i\alpha \in H(v)$. Hence, by the inductive assumption, $\Diamond_i\alpha \in H(w)$.

$\Box$

**Proof of Lemma 4.2** By the definition, $M$ is an $L$-model. Let $M' = \langle W, \tau, R_1, \dots, R_m, H\rangle$ be the extended $L$-model graph of $I$. It can be proved by induction on the length of $\alpha$ that for any $w \in W$, if $\alpha \in H(w)$, then $M', \sigma, w \vDash \alpha$. The cases when $\alpha$ is a classical atom or $\alpha = \langle E\rangle_i\beta$ are trivial. The case when $\alpha = \Box_i\beta$ is solved by Lemma 5.4. Hence $M, \sigma \vDash I$.

**Proof of Theorem 4.3** Let $M = \langle W, \tau, R_1, \dots, R_m, H\rangle$ be the standard $L$-model graph of $I$, $\sigma$ the standard $\Diamond$-realization function and $\{R'_i \mid 1 \leq i \leq m\}$ the skeleton of the standard $L$-model of $I$. By Lemma 4.2, $M$ is an $L$-model of $I$. Let $N = \langle D, W_2, \tau_2, S_1, \dots, S_m, \pi\rangle$ be an arbitrary $L$-model of $I$ and $\sigma_2$ a $\Diamond$-realization function on $N$ such that $N, \sigma_2 \vDash I \cup Serial_L$.

Let $r \subseteq W \times W_2$ be the least relation such that, for all $w$, $w_2$, $u_2$, $E$, $i$:

- $r(\tau, \tau_2)$;
- if $r(w, w_2)$ and $R'_i(w, w\langle E\rangle_i)$ hold, and $\sigma_2(w_2, \langle E\rangle_i)$ is defined, then $r(w\langle E\rangle_i, \sigma_2(w_2, \langle E\rangle_i))$;
- if $r(w, w_2)$ and $S_i(w_2, u_2)$ hold, then $r(w\langle\top\rangle_i, u_2)$.

Note that if $r(w, w_2)$ and $S_i(w_2, u_2)$ hold, then for $u = w\langle\top\rangle_i$ we have $r(u, u_2)$ and $R_i(w, u)$.

We prove that $M \leq_r N$. We first show that if $r(u, u_2)$ and $\alpha \in H(u)$ then $N, \sigma_2, u_2 \vDash \alpha$. We prove this by induction on the length of $u$. Suppose that $r(u, u_2)$ holds and $\alpha \in H(u)$. The case $u = \epsilon$ is trivial. Let $u = w\langle E\rangle_i$ and inductively assume that the assertion holds when $u$ is replaced by $w$. There are two cases:

- $u_2 = \sigma_2(w_2, \langle E\rangle_i)$, $r(w, w_2)$, and $R'_i(w, w\langle E\rangle_i)$, for some $w_2 \in W_2$; or
- $E = \top$, $r(w, w_2)$, and $S_i(w_2, u_2)$, for some $w_2 \in W_2$.

Consider the first case. Since $\alpha \in H(u)$, either $\Box_i\alpha \in H(w)$ or $\langle E\rangle_i\alpha \in H(w)$. By the inductive assumption, either $N, \sigma_2, w_2 \vDash \Box_i\alpha$ or $N, \sigma_2, w_2 \vDash \langle E\rangle_i\alpha$. Hence, $N, \sigma_2, \sigma_2(w_2, \langle E\rangle_i) \vDash \alpha$, which means that $N, \sigma_2, u_2 \vDash \alpha$.

Consider the second case. Since $\alpha \in H(u)$, it follows that $\Box_i\alpha \in H(w)$. By the inductive assumption, $N, \sigma_2, w_2 \vDash \Box_i\alpha$, and hence $N, \sigma_2, u_2 \vDash \alpha$ since $S_i(w_2, u_2)$.

We now show that if $r(w, w_2)$ and $R'_i(w, w\langle E\rangle_i)$ hold then $\sigma_2(w_2, \langle E\rangle_i)$ is defined. The case $E = \top$ is trivial. Suppose that $r(w, w_2)$ and $R'_i(w, w\langle E\rangle_i)$ hold, and $E \neq \top$. Thus, there exists $\langle E\rangle_i\alpha \in H(w)$ for some $\alpha$. Hence $N, \sigma_2, w_2 \vDash \langle E\rangle_i\alpha$ and $\sigma_2(w_2, \langle E\rangle_i)$ is defined. Therefore, the second condition in the above definition of $r$ can be simplified to "if $r(w, w_2)$ and $R'_i(w, w\langle E\rangle_i)$ hold then $r(w\langle E\rangle_i, \sigma_2(w_2, \langle E\rangle_i))$".

It is straightforward to prove by induction on the number of steps needed to obtain $R_i(w, u)$ when extending $\{R'_j \mid 1 \leq j \leq m\}$ to $\{R_j \mid 1 \leq j \leq m\}$ that if $R_i(w, u)$ then: i) if $r(w, w_2)$ then there exists $u_2$ such that $r(u, u_2)$ and $S_i(w_2, u_2)$; ii) if $r(u, u_2)$ then there exists $w_2$ such that $r(w, w_2)$ and $S_i(w_2, u_2)$. We give here only the base case, when $u = w\langle E\rangle_i$ : i) Suppose that $r(w, w_2)$ holds. We have $R'_i(w, w\langle E\rangle_i)$, hence $\sigma_2(w_2, \langle E\rangle_i)$ is defined. The assertion holds for $u_2 = \sigma_2(w_2, \langle E\rangle_i)$. ii) Suppose that $r(u, u_2)$ holds. By the definition of $r$, there exists $w_2$ such that $r(w, w_2)$ and ($S_i(w_2, u_2)$ or $u_2 = \sigma_2(w_2, \langle E\rangle_i)$). It is clear that the assertion holds for such $w_2$.

We have proved that $r$ satisfies all the conditions to guarantee $M \leq_r N$. This together with Lemma 4.2 implies that $M$ is a least $L$-model of $I$.

**Proof of Lemma 4.5** Let $\langle W, \tau, R_1, \ldots, R_m, H\rangle$ be the extended $L$-model graph of $I$, $\triangle$ a modality of the form $\Box_{i_1} \ldots \Box_{i_k}$, and $w = \langle\top\rangle_{i_1} \ldots \langle\top\rangle_{i_k}$. Suppose that $\alpha$ is of the form $\triangle E$. Since $M \vDash \alpha$, we have $M, w \vDash E$. Hence, by Lemma 5.3, $\triangle E \in Sat_L(I)$. Now suppose that $\alpha$ is of the form $\triangle\Diamond_i E$. Since $M \vDash \alpha$, we have $M, w \vDash \Diamond_i E$. By Lemma 5.4, $\Diamond_i E \in H(w)$. Hence, by Lemma 5.3, $\triangle\Diamond_i E \in Sat_L(I)$.

**Proof of Lemma 4.7** Let $M = \langle D, W, \tau, R_1, \ldots, R_m, \pi\rangle$ be an arbitrary $L$-model of $P$ and $\sigma$ a maximal $\Diamond$-realization function on $M$. It is straightforward to prove by induction on $n$ that $M, \sigma \vDash T_{L,P} \uparrow n$. In fact, if $M, \sigma \vDash T_{L,P} \uparrow n$, then $M, \sigma \vDash Sat_L(T_{L,P} \uparrow n)$, and hence $M, \sigma \vDash T_{0L,P}(Sat_L(T_{L,P} \uparrow n))$. Since $NF_L(I) = I$ for any $I$, it follows that $M, \sigma \vDash T_{L,P}(T_{L,P} \uparrow n)$. Therefore $M, \sigma \vDash I_{L,P}$.

**Proof of Lemma 4.8** Let $I'$ be the least extension of $I$ such that, if $\triangle\phi$ is a program clause of $P$, $\phi = (A \leftarrow B_1, \ldots, B_n)$, and $\psi$ is a ground instance of $\phi$, then $\triangle p_\psi \in I'$, where $p_\psi$ is a fresh 0-ary predicate symbol. Let $M$ and $M'$ be the extended $L$-model graphs of $I$ and $I'$, respectively. Clearly, these model graphs have the same frame. Let $M = \langle W, \tau, R_1, \ldots, R_m, H\rangle$ and $M' = \langle W, \tau, R_1, \ldots, R_m, H'\rangle$.

Let $\triangle\phi$ be a program clause of $P$, $\phi = (A \leftarrow B_1, \ldots, B_n)$, and $\psi$ a ground instance of $\phi$. By Lemmas 4.2 and 5.2, $M' \vDash \triangle p_\psi$. To prove that $M \vDash P$ it is sufficient to show that for any $w \in W$, if $p_\psi \in H'(w)$ then $M, w \vDash \psi$. Suppose that $p_\psi \in H'(w)$.

Let $\triangle'' = w$ and $\triangle' = \Box_{i_1} \ldots \Box_{i_k}$ be a $\Box$-lifting form of $\triangle''$ . By Lemma 5.3, some $\Box$-lifting form of $\triangle'' p_\psi$ belongs to $Sat_L(I')$. This $\Box$-lifting form must be $\triangle' p_\psi$. Thus $\triangle' p_\psi \in Sat_L(\{\triangle p_\psi\})$. Hence $\triangle p_\psi \to \triangle' p_\psi$ is $L$-valid and the program clause $\triangle'\psi$ is a ground $L$-instance of $\triangle\phi$.

Let $\psi = (A' \leftarrow B'_1, \ldots, B'_n)$ and suppose that $M, w \vDash B'_i$ for all $1 \leq i \leq n$. We show that $M, w \vDash A'$. If $B'_i$ is of the form $\Box_j E$, then by Lemma 5.3, some $\Box$-lifting form of $\triangle''\langle\top\rangle_j E$ belongs to $Sat_L(I)$. If $B'_i$ is of the form $\Diamond_j E$, then by Lemma 5.4, $\Diamond_j E \in H(w)$. Hence, by Lemma 5.3, some $\Box$-lifting form of $\triangle'' B'_i$ belongs to $Sat_L(I)$. Consequently, some $\Box$-lifting form $\alpha$ of $\triangle'' A''$, where $A''$ is the forward labeled form of $A'$, belongs to $T_{0L,P}(Sat_L(I))$. Since $T_{0L,P}(Sat_L(I)) = T_{L,P}(I) \subseteq I$, by Lemma 4.2, we have that $M, \sigma \vDash \alpha$, where $\sigma$ is the standard $\Diamond$-realization function on $M$. Hence $M, w \vDash A'$. We have shown that $M, w \vDash \psi$ and completed the proof.

# 6 Programming about Multi-degree Belief

To reason about multi-degree belief we can use the multimodal logics $KDI4$, $KDI4_s$, $KDI4_s5$, and $KDI45$. Recall that, in these logics, $\Box_i\phi$ stands for "$\phi$ is believed at degree $i$" and $\Diamond_i\phi$ stands for "it is possible weakly at degree $i$ that $\phi$". In this section, we give schemata for semantics of MProlog in the mentioned logics and prove their correctness. We also present an example in the last subsection.

| $L = KDI4_s$, $L$-MProlog | | |
|---|---|---|
| $\preceq_L$ is defined by Definition 4.8. | | |
| No restrictions on the $L$-normal form of modalities.<br>No rules specifying $NF_L$ and $rNF_L$. | | |
| Rules specifying | | |
| $Ext_L$ | $\triangle\Box_i\alpha \to \triangle\Box_j\alpha$ if $i > j$ | (1) |
| | $\triangle\Box_i\alpha \to \triangle\Box_j\Box_i\alpha$ | (2) |
| $Sat_L$ | the rules specifying $Ext_L$ plus | |
| | $\triangle\nabla\nabla' E \to \triangle\Diamond_i E$ if $\Diamond_i \preceq_L \nabla'$ | (3) |
| $rSat_L$ | $\triangle\Diamond_i E \leftarrow \triangle\langle X\rangle_i E$ for $X$ being a fresh atom variable | (4) |
| | $\triangle\nabla_i\alpha \leftarrow \triangle\Box_j\alpha$ if $i \leq j$ | (5) |
| | $\triangle\Diamond_i E \leftarrow \triangle\Diamond_j E$ if $i > j$ | (6) |
| | $\triangle\nabla\Box_i\alpha \leftarrow \triangle\Box_i\alpha$ | (7) |
| | $\triangle\Diamond_i E \leftarrow \triangle\langle X\rangle_j\Diamond_i E$ for $X$ being a fresh atom variable | (8) |

Table 2: A schema for semantics of $KDI4_s$-MProlog

## 6.1 Schemata for Semantics of MProlog in $KDI4_s$ and $KDI4$

In this subsection, let $L$ denote either $KDI4_s$ or $KDI4$. In Table 2, we give a schema for semantics of $KDI4_s$-MProlog. The rule (1) in that schema follows from the axiom $(I)$; the rules (2) and (7) are based on the axiom $(4_s)$; the rules (3) and (8) are based on the reverse of the axiom $(4_s)$, i.e. $\Diamond_j\Diamond_i\phi \to \Diamond_i\phi$; the rule (5) is based on the axioms $(D)$ and $(I)$; and finally, the rule (6) follows from the reverse of the axiom $(I)$, i.e. $\Diamond_j\phi \to \Diamond_i\phi$ if $i > j$.

In Table 3, we give a schema for semantics of $KDI4$-MProlog. The rule (1) in that schema follows from the axiom $(I)$; the rules (2) and (7) are based on the axiom (4); the rules (3) and (8) are based on the reverse of the axiom (4), i.e. $\Diamond_i\Diamond_i\phi \to \Diamond_i\phi$; the rule (5) is based on the axioms $(D)$ and $(I)$; and finally, the rule (6) follows from the reverse of the axiom $(I)$.

The schemata given in Tables 2 and 3 are formulated so that they can use the proofs given in the Section 4. However, the SLD-resolution calculi can be modified to be more effective. When resolving a goal with an input clause, if we relax the condition that the mgu $\theta$ unifies the selected head atom $A'$ with the forward labeled form $A''$ of the head of the input clause, but only require that $\theta$ is a most general substitution such that $A'\theta$, $A''\theta$ have the same classical atom and $A'\theta$ is an $L$-instance of $A''\theta$, then the rules (5) and (7) in both of the schemata can be combined and replaced by: $\triangle\nabla\nabla'_i\alpha \leftarrow \triangle\Box_i\alpha$ for $KDI4_s$; and $\triangle\nabla_i\nabla'_j\alpha \leftarrow \triangle\Box_k\alpha$, where $k = max(i,j)$, for $KDI4$. It can be shown that every SLD-refutation in one of the original calculi can be simulated in the new corresponding calculus by another one with the same computed answer. This means that the new SLD-resolution calculi are sound and complete, provided that so are the original calculi.

**Theorem 6.1** *The schemata for semantics of $KDI4_s$-MProlog and $KDI4$-MProlog are correct.*

The proof of this theorem is given in Appendix B.

## 6.2 A Schema for Semantics of $KDI4_s5$-MProlog

In this subsection, let $L$ denote the logic $KDI4_s5$. It can be checked that a connected frame $\langle W, \tau, R_1, \ldots, R_m\rangle$ is a $KDI4_s5$-frame iff there are nonempty subsets of worlds $W_1 \subseteq \ldots \subseteq W_m$ such that $W = \{\tau\} \cup W_m$ and $R_i = W \times W_i$, for $1 \leq i \leq m$.

| $L = KDI4$, $L$-MProlog | | |
|---|---|---|
| $\preceq_L$ is defined by Definition 4.8. | | |
| No restrictions on the $L$-normal form of modalities.<br>No rules specifying $NF_L$ and $rNF_L$. | | |
| Rules specifying | | |
| $Ext_L$ | $\triangle\Box_i\alpha \to \triangle\Box_j\alpha$ if $i > j$ | (1) |
| | $\triangle\Box_i\alpha \to \triangle\Box_i\Box_i\alpha$ | (2) |
| $Sat_L$ | the rules specifying $Ext_L$ plus | |
| | $\triangle\nabla\nabla' E \to \triangle\Diamond_i E$ if $\Diamond_i \preceq_L \nabla$ and $\Diamond_i \preceq_L \nabla'$ | (3) |
| $rSat_L$ | $\triangle\Diamond_i E \leftarrow \triangle\langle X\rangle_i E$ for $X$ being a fresh atom variable | (4) |
| | $\triangle\nabla_i\alpha \leftarrow \triangle\Box_j\alpha$ if $i \leq j$ | (5) |
| | $\triangle\Diamond_i E \leftarrow \triangle\Diamond_j E$ if $i > j$ | (6) |
| | $\triangle\Box_i\Box_i\alpha \leftarrow \triangle\Box_i\alpha$ | (7) |
| | $\triangle\Diamond_i E \leftarrow \triangle\langle X\rangle_j\Diamond_i E$ for $X$ being a fresh atom variable and $i \geq j$ | (8) |

Table 3: A schema for semantics of $KDI4$-MProlog

In Table 4, we give a schema for semantics of $KDI4_s5$-MProlog. The $L$-normal form of modalities and the rules (2)–(5) and (9) in that schema are justified by the $L$-tautology $\nabla\phi \equiv \nabla'\nabla\phi$, where $\nabla$ and $\nabla'$ are unlabeled modal operators. The rule (1) follows from the axiom $(I)$, the rule (7) is based on the axioms $(D)$ and $(I)$, and the rule (8) follows from the reverse of the axiom $(I)$.

The schema given in Table 4 is formulated so that it can use the proofs given in the Section 4. However, the rules (6), (7), (8) of Table 4 can be simplified by deleting the occurrences of $\triangle$ and replacing $\alpha$ by $E$. Furthermore, when resolving a goal with an input clause, if we relax the condition that the mgu $\theta$ unifies the selected head atom $A'$ with the forward labeled form $A''$ of the head of the input clause, but only require that $\theta$ is a most general substitution such that $A'\theta$, $A''\theta$ have the same classical atom and $A'\theta$ is an $L$-instance of $A''\theta$, then the rule (7) can be deleted. It can be shown that every SLD-refutation in the original calculus can be simulated in the new calculus by another one with the same or a more general computed answer. This means that the new SLD-resolution calculus is sound and complete, provided that so is the original calculus.

**Theorem 6.2** *The schema for semantics of $KDI4_s5$-MProlog is correct.*

We now prove this theorem. Lemmas 4.4 and 4.10 – 4.13 can be easily verified.

**Proof of Lemma 4.2** By the definition, $M$ is an $L$-model. Let $\{R'_i \mid 1 \leq i \leq m\}$ be the skeleton of $M$. We prove by induction on the length of $\alpha$ that for any $w \in W$, if $\alpha \in H(w)$, then $M, \sigma, w \models \alpha$. The cases when $\alpha$ is a classical atom or $\alpha = \langle E\rangle_i F$ (and $w = \tau$) are trivial. Consider the remaining case when $\alpha = \Box_i E$ and $w = \tau$. Let $u$ be a world such that $R_i(\tau, u)$ holds. We show that $E \in H(u)$. Since $R_i(\tau, u)$, $u$ must be of the form $\langle F\rangle_j$ for some $F$ and $j \leq i$. Since $\Box_i E \in H(\tau)$, by the definition of $Ext_L$, we have $\Box_j E \in H(\tau)$, and hence $E \in H(u)$.

**Proof of Theorem 4.3** Let $M = \langle W, \tau, R_1, \ldots, R_m, H\rangle$ be the standard $L$-model graph of $I$, $\sigma$ the standard $\Diamond$-realization function and $\{R'_i \mid 1 \leq i \leq m\}$ the skeleton of the standard $L$-model of $I$. By Lemma 4.2, $M$ is an $L$-model of $I$. Let $N = \langle D, W_2, \tau_2, S_1, \ldots, S_m, \pi\rangle$ be an arbitrary $L$-model of $I$ and $\sigma_2$ a $\Diamond$-realization function on $N$ such that $N, \sigma_2 \models I \cup Serial_L$.

We first show that if $R'_i(\tau, \langle E\rangle_i)$ holds then $\sigma_2(\tau_2, \langle E\rangle_i)$ is defined. The case $E = \top$ is trivial. Suppose that $R'_i(\tau, \langle E\rangle_i)$ holds and $E \neq \top$. Thus, there exists $\langle E\rangle_i\alpha \in H(\tau)$ for some $\alpha$. Hence

| $L = KDI4_s5$, $\quad L$-MProlog | | |
|---|---|---|
| $\preceq_L$ is defined by Definition 4.8. | | |
| A modality is in the $L$-normal form if its length $\leq 1$. | | |
| Rules specifying | | |
| $Ext_L$ | $\Box_i E \to \Box_j E$ if $i > j$ | (1) |
| $Sat_L$ | the rules specifying $Ext_L$ plus | |
| | $\Box_i E \to \Box_m \Box_i E$ | (2) |
| | $\langle F \rangle_i E \to \Box_m \Diamond_i E$ | (3) |
| $NF_L$ | $\nabla\nabla' E \to \nabla' E$ if $\nabla'$ is of the form $\Box_i$ or $\langle E \rangle_i$ | (4) |
| $rNF_L$ | $\nabla E \leftarrow \langle X \rangle_j \nabla E$ if $\nabla$ is of the form $\Box_i$ or $\langle E \rangle_i$ | |
| | and $X$ is a fresh atom variable | (5) |
| $rSat_L$ | $\triangle \Diamond_i E \leftarrow \triangle \langle X \rangle_i E$ for $X$ being a fresh atom variable | (6) |
| | $\triangle \nabla_i \alpha \leftarrow \triangle \Box_j \alpha$ if $i \leq j$ | (7) |
| | $\triangle \Diamond_i E \leftarrow \triangle \Diamond_j E$ if $i > j$ | (8) |
| | $\nabla\nabla' E \leftarrow \nabla' E$ if $\nabla'$ is of the form $\Box_i$ or $\Diamond_i$ | (9) |

Table 4: A schema for semantics of $KDI4_s5$-MProlog

$N, \sigma_2, \tau_2 \vDash \langle E \rangle_i \alpha$, and $\sigma_2(w_2, \langle E \rangle_i)$ is defined.

Let $r \subseteq W \times W_2$ be the least relation such that, for all $w$, $w_2$, $u_2$, $E$, $i$:

- $r(\tau, \tau_2)$;
- if $R'_i(\tau, \langle E \rangle_i)$ holds then $r(\langle E \rangle_i, \sigma_2(\tau_2, \langle E \rangle_i))$;
- if $r(w, w_2)$ and $S_i(w_2, u_2)$ hold, then $r(\langle \top \rangle_i, u_2)$.

We prove that $M \leq_r N$. If $r(w, w_2)$ and $S_i(w_2, u_2)$ hold, then for $u = \langle \top \rangle_i$ we have $r(u, u_2)$ and $R_i(w, u)$. We proceed by showing that if $r(u, u_2)$ and $\alpha \in H(u)$ then $N, \sigma_2, u_2 \vDash \alpha$. The case $u = \tau$ is trivial. Suppose that $u = \langle E \rangle_i$, $r(u, u_2)$, and $\alpha \in H(u)$. There are two cases:

- $u_2 = \sigma_2(\tau_2, \langle E \rangle_i)$ and $R'_i(\tau, \langle E \rangle_i)$; or
- $E = \top$, $r(w, w_2)$, and $S_i(w_2, u_2)$, for some $w$, $w_2$.

Consider the first case. Since $\alpha \in H(u)$, either $\Box_i \alpha \in H(\tau)$ or $\langle E \rangle_i \alpha \in H(\tau)$. Hence, $N, \sigma_2, \tau_2 \vDash \Box_i \alpha$ or $N, \sigma_2, \tau_2 \vDash \langle E \rangle_i \alpha$. It follows that $N, \sigma_2, u_2 \vDash \alpha$.

Consider the second case. Since $\alpha \in H(u)$, it follows that $\Box_i \alpha \in H(\tau)$. Hence, $N, \sigma_2, \tau_2 \vDash \Box_i \alpha$. Since $r(w, w_2)$ and $S_i(w_2, u_2)$, it can be shown that $u_2$ is directly or indirectly reachable from $\tau_2$ (via the accessibility relations $S_j$, $1 \leq j \leq m$). Hence $S_i(\tau_2, u_2)$ holds, and $N, \sigma_2, u_2 \vDash \alpha$.

To prove $M \leq_r N$, it remains to show that if $r(w, w_2)$ and $R_i(w, u)$ hold, then there exists $u_2 \in W_2$ such that $r(u, u_2)$ and $S_i(w_2, u_2)$ hold. Suppose that $r(w, w_2)$ and $R_i(w, u)$ hold. It follows that $R'_j(\tau, u)$ holds for some $j \leq i$. Let $u = \langle E \rangle_j$ and choose $u_2 = \sigma_2(\tau_2, \langle E \rangle_j)$. Thus we have $r(u, u_2)$. Since $r(w, w_2)$, it can be shown that $w_2$ is directly or indirectly reachable from $\tau_2$ (via the accessibility relations $S_j$, $1 \leq j \leq m$). Hence $S_i(w_2, u_2)$ holds.

**Proof of Lemma 4.5** If $\alpha$ is of the form $E$ or $\Box_i E$, then $\alpha \in Ext_L(I)$ (since $M \vDash \alpha$), and hence $\alpha \in Sat_L(I)$. Suppose that $\alpha = \Diamond_i E$. Let $\langle W, \tau, R_1, \ldots, R_m, H \rangle$ be the standard $L$-model graph of

$I$. Since $M \vDash \alpha$, there exists a world $u = \langle F \rangle_j$ of $M$ such that $j \leq i$ and $E \in H(u)$. By Lemma 4.1, some $\Box$-lifting form of $\langle F \rangle_j E$ belongs to $Ext_L(I)$. It follows that either $\Box_j E$ or $\langle F \rangle_j E$ belongs to $Ext_L(I)$. Hence $\alpha$ is an $L$-instance of some atom from $Sat_L(I)$.

**Proof of Lemma 4.7** Let $M = \langle D, W, \tau, R_1, \ldots, R_m, \pi \rangle$ be an arbitrary $L$-model of $P$ and $\sigma$ a maximal $\Diamond$-realization function on $M$. Note that if $M, \sigma \vDash \nabla \langle E \rangle_i E$ then $M, \sigma \vDash \langle E \rangle_i E$. It is straightforward to prove by induction on $n$ that $M, \sigma \vDash T_{L,P} \uparrow n$. Therefore $M, \sigma \vDash I_{L,P}$ and $P \vDash_L I_{L,P}$.

**Proof of Lemma 4.8** Let $M$ be the standard $L$-model of $I$ and $\sigma$ the standard $\Diamond$-realization function on $M$. It is sufficient to prove that for any ground $L$-instance $\triangle(A \leftarrow B_1, \ldots, B_n)$ of some program clause of $P$, for any $w \in W$ being an $L$-instance of $\triangle$, $M, w \vDash (A \leftarrow B_1, \ldots, B_n)$. Suppose that $M, w \vDash B_i$ for all $1 \leq i \leq n$. We show that $M, w \vDash A$.

Let $\triangle' = w$. We first show that for any ground simple atom $B$ of the form $E$, $\Box_i E$, or $\Diamond_i E$, if $M, w \vDash B$ then $\triangle' B$ is an $L$-instance of some atom from $Sat_L(I)$. Suppose that $M, w \vDash B$. If $B$ is of the form $E$, then by Lemma 4.1, some $\Box$-lifting form of $\triangle' B$ belongs to $Ext_L(I)$, and hence $\triangle' B$ is an $L$-instance of some atom from $Sat_L(I)$. If $B$ is of the form $\Box_i E$ then, by the construction of $M$, it follows that $\Box_i E \in Ext_L(I)$, and hence $\{\Box_i E, \Box_m \Box_i E\} \subseteq Sat_L(I)$, which implies that $\triangle' B$ is an $L$-instance of some atom from $Sat_L(I)$. Now consider the case when $B$ is of the form $\Diamond_i E$. Since $M, w \vDash \Diamond_i E$, either $\Box_j E \in Ext_L(I)$ or $\langle F \rangle_j E \in Ext_L(I)$ for some $F$ and $j \leq i$. Hence, either $\{\Box_j E, \Box_m \Box_j E\} \subseteq Sat_L(I)$ or $\{\langle F \rangle_j E, \Box_m \Diamond_j E\} \subseteq Sat_L(I)$ for some $F$ and $j \leq i$. Therefore $\triangle' B$ is an $L$-instance of some atom from $Sat_L(I)$.

Since $M, w \vDash B_i$ for $1 \leq i \leq n$, it follows that $\triangle' B_i$ is an $L$-instance of some atom from $Sat_L(I)$. Consequently, $\triangle' A$ is an $L$-instance of some atom $\alpha$ from $T_{0L,P}(Sat_L(I))$. Suppose that $\alpha$ is in the $L$-normal form. We have $\alpha \in T_{L,P}(I) \subseteq I$. By Lemma 4.2, we have that $M, \sigma \vDash \alpha$, and hence $M, w \vDash A$. Now suppose that $\alpha$ is not in the $L$-normal, i.e. $mdepth(\alpha) = 2$. Thus $A$ is of the form $\Box_i E$ or $\Diamond_i E$. Let $A'$ be the forward labeled form of $A$. Since $A' \in T_{L,P}(I)$, by Lemma 4.2, $M, \sigma \vDash A'$. Hence $M, w \vDash A$.

### 6.3 A Schema for Semantics of $KDI45$-MProlog

In this section, let $L$ denote the logic $KDI45$. We first show that $\nabla_i \nabla'_j \alpha \equiv \nabla'_j \alpha$ is $L$-valid for $i \leq j$, where $\nabla_i$ and $\nabla'_j$ are unlabeled modal operators. Suppose that $i \leq j$. There are the following cases:

- $\nabla_i = \Box_i$ and $\nabla'_j = \Box_j$ : $\Box_i \Box_j \alpha \to \Box_j \alpha$ follows from $\Box_i \Box_j \alpha \to \Diamond_i \Box_j \alpha$, $\Diamond_i \Box_j \alpha \to \Diamond_j \Box_j \alpha$, and $\Diamond_j \Box_j \alpha \to \Box_j \alpha$. For the conversion, $\Box_j \alpha \to \Box_i \Box_j \alpha$ follows from $\Box_j \alpha \to \Box_j \Box_j \alpha$ and $\Box_j \Box_j \alpha \to \Box_i \Box_j \alpha$.
- $\nabla_i = \Box_i$ and $\nabla'_j = \Diamond_j$ : $\Box_i \Diamond_j \alpha \to \Diamond_j \alpha$ follows from $\Box_i \Diamond_j \alpha \to \Diamond_i \Diamond_j \alpha$, $\Diamond_i \Diamond_j \alpha \to \Diamond_j \Diamond_j \alpha$, and $\Diamond_j \Diamond_j \alpha \to \Diamond_j \alpha$. For the conversion, $\Diamond_j \alpha \to \Box_i \Diamond_j \alpha$ follows from $\Diamond_j \alpha \to \Box_j \Diamond_j \alpha$ and $\Box_j \Diamond_j \alpha \to \Box_i \Diamond_j \alpha$.
- $\nabla_i = \Diamond_i$ and ($\nabla'_j = \Box_j$ or $\nabla'_j = \Diamond_j$) : These cases are dual to the above cases.

The above observation suggests that a modality $\nabla^{(1)}_{i_1} \ldots \nabla^{(k)}_{i_k}$ is in the $L$-normal form if $i_1 > \ldots > i_k$.

In Table 5, we give a schema for semantics of $KDI45$-MProlog. The rule (1) follows from the axiom (I); the rule (2) from the axioms (4) and (I); the rule (3) from the axioms (I), (D), and (5) (because $\triangle \Box_i \Box_j \alpha \to \triangle \Box_j \Box_j \alpha$, $\triangle \Box_j \Box_j \alpha \to \triangle \Diamond_j \Box_j \alpha$, and $\triangle \Diamond_j \Box_j \alpha \to \triangle \Box_j \alpha$); the rule (4) follows from the axiom (4); the rule (5) is based on the axiom (5); the rules (6) and (7) are based on the axiom (4) (as $\triangle \Diamond_k \Diamond_k E \to \triangle \Diamond_k E$); the rules (8) and (9) are based on the mentioned observation on the $L$-normal form of modalities; the rule (11) is based on the axioms ($D$) and ($I$); the rule (12) is based on the axiom (I); the rule (13) is dual to the rules (2) and (4); the rule (14) is dual to the rule (5); the rule (15) is dual to the rule (3); and finally, the rule (16) is based on the axiom (4) and dual to the rules (6) and (7).

Note that if $\nabla_i$ is a maximally general $L$-instance of labeled modal operators $\nabla'_j$ and $\nabla''_k$ then $i = min(j, k)$. It follows that if $\triangle$ is a maximally general $L$-instance of $\triangle^{(1)}, \ldots, \triangle^{(k)}$, which are modalities in the $L$-normal labeled form, then $\triangle$ is also in the $L$-normal labeled form.

| $L = KDI45$, $L$-MProlog |
|---|
| $\preceq_L$ is defined by Definition 4.8.<br><br>$\nabla^{(1)}_{i_1} \ldots \nabla^{(k)}_{i_k}$ is in the $L$-normal form if $i_1 > \ldots > i_k$.<br><br>The following rules are accompanied with the condition that the atoms in both sides are in the $L$-normal labeled form for rules specifying $Ext_L$ and in the almost $L$-normal labeled form for other rules. (*)<br><br>$Ext_L$ $\triangle\Box_i\alpha \to \triangle\Box_j\alpha$ if $i > j$ (1)<br>$\triangle\Box_i\alpha \to \triangle\Box_i\Box_j\alpha$ if $i > j$ (2)<br>$\triangle\Box_i\Box_j\alpha \to \triangle\Box_j\alpha$ if $i > j$ (3)<br><br>$Sat_L$ the rules for $Ext_L$ with the modification stated in (*), plus<br>$\triangle\Box_i E \to \triangle\Box_i\Box_i E$ (4)<br>$\triangle\nabla E \to \triangle\Box_i\Diamond_i E$ if $\Diamond_i \preceq_L \nabla$ (5)<br>$\triangle\Box_i\nabla_j E \to \triangle\Diamond_j E$ if $i > j$ (6)<br>$\triangle\langle F\rangle_i\nabla_j E \to \triangle\Diamond_i E$ if $i > j$ (7)<br><br>$NF_L$ $\triangle\nabla_i\nabla'_j E \to \triangle\nabla'_j E$ if $\nabla'_j$ is of the form $\Box_j$ or $\langle E\rangle_j$ and $i \leq j$ (8)<br><br>$rNF_L$ $\triangle\nabla_j E \leftarrow \triangle\langle X\rangle_i\nabla_j E$ if $\nabla_j$ is of the form $\Box_j$ or $\langle E\rangle_j$, $X$ is a fresh atom variable, and $i \leq j$ (9)<br><br>$rSat_L$ $\triangle\Diamond_i E \leftarrow \triangle\langle X\rangle_i E$ for $X$ being a fresh atom variable (10)<br>$\triangle\nabla_i\alpha \leftarrow \triangle\Box_j\alpha$ if $i \leq j$ (11)<br>$\triangle\Diamond_i E \leftarrow \triangle\Diamond_j E$ if $i > j$ (12)<br>$\triangle\Box_i\Box_j\alpha \leftarrow \triangle\Box_i\alpha$ if $i \geq j$ (13)<br>$\triangle\Box_i\Diamond_i E \leftarrow \triangle\Diamond_i E$ (14)<br>$\triangle\Box_i\alpha \leftarrow \triangle\Box_j\Box_i\alpha$ if $i < j$ (15)<br>$\triangle\Diamond_i E \leftarrow \triangle\langle X\rangle_i\Diamond_i E$ for $X$ being a fresh atom variable (16) |

Table 5: A schema for semantics of $KDI45$-MProlog

**Theorem 6.3** *The schema for semantics of $KDI4_s5$-MProlog is correct.*

The proof of this theorem is given in Appendix C.

### 6.4 Examples

**Example 6.1** Assume that there are 5 degrees of belief. Consider the program $P$ consisting of the following clauses:

$$\begin{aligned}
\phi_1 &= \Box_4 good_in_maths(x) \leftarrow maths_teacher(x)\\
\phi_2 &= \Box_5(\Box_i good_in_maths(x) \leftarrow \Box_i mathematician(x))\\
\phi_3 &= \Box_3(\Diamond_i good_in_maths(x) \leftarrow maths_student(x))\\
\phi_4 &= \Box_3(\Diamond_i good_in_physics(x) \leftarrow physics_student(x))\\
\phi_5 &= \Box_2(\Diamond_2 good_in_maths(x) \leftarrow good_in_physics(x))\\
\phi_6 &= maths_teacher(John) \leftarrow\\
\phi_7 &= \Box_2 mathematician(Tom) \leftarrow\\
\phi_8 &= \Box_5 maths_student(Peter) \leftarrow\\
\phi_9 &= \Box_5 physics_student(Mike) \leftarrow
\end{aligned}$$

The index $i$ in the above rules can take any value from the range [1..5]. For the goal $\leftarrow \Box_4 good\_in\_maths(x)$, in all of the logics considered in this section, we have the correct answer $\{x = John\}$. For the goal $\leftarrow \Box_2 good\_in\_maths(x)$, in the logics $KDI4_s5$ and $KDI45$, we have the additional correct answer $\{x = Tom\}$. For the goal $\leftarrow \Diamond_1 good\_in\_maths(x)$, in all of the logics, we have three correct answers $\{x = John\}$, $\{x = Tom\}$, and $\{x = Peter\}$. The goal $\leftarrow \Diamond_2 good\_in\_maths(Mike)$ successes only in the logic $KDI4_s5$. We give here an SLD-refutation of $P \cup \{\leftarrow \Diamond_2 good\_in\_maths(x)\}$ in $KDI4_s5$ with computed answer $\{x/Mike\}$ :

| Goal | Input Clause/Rule, Mgu, Constraints |
|---|---|
| $\leftarrow \Diamond_2 good\_in\_maths(x)$ | |
| $\leftarrow \langle X\rangle_2 good\_in\_maths(x)$ | (6) |
| $\leftarrow \langle Y\rangle_j \langle good\_in\_maths(x)\rangle_2 good\_in\_maths(x)$ | $(5), \{X/good\_in\_maths(x)\}$ |
| $\leftarrow \langle Y\rangle_j good\_in\_physics(x)$ | $\phi_5, \{x_3/x\}, j \leq 2$ |
| $\leftarrow \Box_j good\_in\_physics(x)$ | $(7), \epsilon, j \leq 2$ |
| $\diamond$ | $\phi_9, \{x/Mike\}$ |

In the above refutation, $j$ can take value 1 or 2. In another work, we have implemented MProlog as an additional module to Prolog, and constraints as goal atoms. With that module, we can also consider, for example, the goals $\leftarrow \Box_i good\_in\_maths(x)$ and $\leftarrow \Diamond_i good\_in\_maths(x)$.

## 7 Programming in MProlog for Multi-agent Systems

In this section, we give schemata for semantics of MProlog in the logics $KD4_s5_s$, $KD45_{(m)}$, and $KD4I_g5_a$. In these logics, $\Box_i\phi$ stands for "$\phi$ is believed by agent $i$", while $\Diamond_i\phi$ stands for "$\phi$ is considered possible by agent $i$". The logic $KD4_s5_s$ can be used for distributed systems of belief, in which agents have full access to belief bases of each other. The logics $KD45_{(m)}$ and $KD4I_g5_a$ are intended for reasoning about epistemic states of agents. In $KD4I_g5_a$, some modal indexes stand for groups of agents, and using them we can reason about common belief.

### 7.1 A Schema for Semantics of $KD4_s5_s$-MProlog

In this subsection, let $L$ denote $KDI4_s5$. It can be checked that a connected frame $\langle W, \tau, R_1, \ldots, R_m\rangle$ is a $KD4_s5_s$-frame iff there are nonempty subsets of worlds $W_1, \ldots, W_m$ such that $W = \{\tau\} \cup W_1 \cup \ldots \cup W_m$ and $R_i = W \times W_i$, for $1 \leq i \leq m$. Note that this property is similar to the $KDI4_s5$-frame restrictions. The difference is that the logic $KD4_s5_s$ does not contain the axiom $(I)$ and in this logic we do not have the condition that $W_i \subseteq W_j$ for $i < j$.

In Table 6, we give a schema for semantics of $KD4_s5_s$-MProlog. The $L$-normal form of modalities and the rules (1)–(4) and (7) in that schema are justified by the $L$-tautology $\nabla\phi \equiv \nabla'\nabla\phi$, where $\nabla$

| $L = KD4_s5_s$, $L$-MProlog | | |
|---|---|---|
| $\preceq_L$ is defined by Definition 4.8. | | |
| A modality is in the $L$-normal form if its length $\leq 1$. | | |
| Rules specifying | | |
| $Ext_L$ | no rules | |
| $Sat_L$ | $\Box_i E \to \Box_j \Box_i E$ | (1) |
| | $\langle F \rangle_i E \to \Box_j \Diamond_i E$ | (2) |
| $NF_L$ | $\nabla\nabla' E \to \nabla' E$ if $\nabla'$ is of the form $\Box_i$ or $\langle E \rangle_i$ | (3) |
| $rNF_L$ | $\nabla E \leftarrow \langle X \rangle_j \nabla E$ if $\nabla$ is of the form $\Box_i$ or $\langle E \rangle_i$ and $X$ is a fresh atom variable | (4) |
| $rSat_L$ | $\triangle \Diamond_i E \leftarrow \triangle \langle X \rangle_i E$ for $X$ being a fresh atom variable | (5) |
| | $\triangle \nabla_i \alpha \leftarrow \triangle \Box_i \alpha$ | (6) |
| | $\nabla\nabla' E \leftarrow \nabla' E$ if $\nabla'$ is of the form $\Box_i$ or $\Diamond_i$ | (7) |

Table 6: A schema for semantics of $KD4_s5_s$-MProlog

and $\nabla'$ are unlabeled modal operators, while the rule (6) is based on the axiom $(D)$. This schema is similar to the schema for semantics of $KDI4_s5$-MProlog, except that it does not contain rules involving with the axiom $(I)$. Analogously as for $KDI4_s5$, we can prove the following theorem.

**Theorem 7.1** *The schema for semantics of $KD4_s5_s$-MProlog is correct.*

The schema given in Table 6 is formulated so that it can use the proofs given in the Section 4. However, similarly as for the case of $KDI4_s5$, the rules (5) and (6) of Table 6 can be simplified in the way that the occurrences of $\triangle$ in those rules are deleted and $\alpha$ in the rule (6) is replaced by $E$. Furthermore, when resolving a goal with an input clause, if we relax the condition that the mgu $\theta$ unifies the selected head atom $A'$ with the forward labeled form $A''$ of the head of the input clause, but only require that $\theta$ is a most general substitution such that $A'\theta$, $A''\theta$ have the same classical atom and $A'\theta$ is an $L$-instance of $A''\theta$, then the rule (6) can be deleted. It can be shown that every SLD-refutation in the original calculus can be simulated in the new calculus by another one with the same computed answer. This means that the new SLD-resolution calculus is also sound and complete.

An agent should keep all clauses that define its epistemic states. This means that agent $i$ should keep all clauses of the form $\nabla_i E \leftarrow B_1, \ldots, B_n$ or $\Box_i(A \leftarrow B_1, \ldots, B_n)$. Furthermore, program clauses of the form $\Box_i(\Box_j E \leftarrow B_1, \ldots, B_n)$ with $i \neq j$ have little sense in this kind of multi-agent systems. It can be shown that program clauses of that form can be disallowed without reducing expressiveness of $KD4_s5_s$-MProlog. If we adopt this restriction then the rule (4) in Table 6 can be modified so that the involved modal operators have the same kind (i.e. agent index). Program clauses of the form $E \leftarrow B_1, \ldots, B_n$ can be kept by a special agent, which will communicate with users. Whenever an agent meets a goal atom of the form $\nabla_i E$ it will require the agent $i$ to solve the goal $\leftarrow \nabla_i E$, and whenever an agent meets a goal atom of the form $E$ (without modal context) it will require the special agent to solve the goal $\leftarrow E$.

**Example 7.1** Let $P$ consist of the following clauses:

**agent 1:**
$\phi_1 = \Box_1 likes(Jan, cola) \leftarrow$
$\phi_2 = \Box_1 likes(Piotr, pepsi) \leftarrow$
$\phi_3 = \Box_1(\Diamond_1 likes(x, cola) \leftarrow likes(x, pepsi))$
$\phi_4 = \Box_1(\Diamond_1 likes(x, pepsi) \leftarrow likes(x, cola))$
**agent 2:**
$\phi_5 = \Box_2 likes(Jan, pepsi) \leftarrow$
$\phi_6 = \Box_2 likes(Piotr, cola) \leftarrow$
$\phi_7 = \Box_2 likes(Piotr, beer) \leftarrow$
$\phi_8 = \Box_2(likes(x, cola) \leftarrow likes(x, pepsi))$
$\phi_9 = \Box_2(likes(x, pepsi) \leftarrow likes(x, cola))$
**agent 3:**
$\phi_{10} = \Box_3 likes(Jan, cola) \leftarrow$
$\phi_{11} = \Diamond_3 likes(Piotr, pepsi) \leftarrow$
$\phi_{12} = \Diamond_3 likes(Piotr, beer) \leftarrow$
$\phi_{13} = \Box_3(very\_much\_likes(x, y) \leftarrow likes(x, y), \Box_1 likes(x, y), \Box_2 likes(x, y))$
**agent communicating with users:**
$\phi_{14} = very\_much\_likes(x, y) \leftarrow \Box_3 very\_much\_likes(x, y)$
$\phi_{15} = likes(x, y) \leftarrow \Diamond_3 very\_much\_likes(x, y)$
$\phi_{16} = possibly\_likes(x, y) \leftarrow \Diamond_i likes(x, y)$

The modal index $i$ in $\phi_{16}$ can take value 1, 2, or 3. Let the base logic be $KD4_s5_s$. For the goal $\leftarrow very\_much\_likes(x, y)$, we have the unique computed answer $\{x/Jan, y/cola\}$. For the goal $\leftarrow likes(x, y)$, we have two computed answers $\{x/Jan, y/cola\}$ and $\{x/Piotr, y/pepsi\}$. For the goal $\leftarrow possibly\_likes(x, y)$ we have 5 computed answers. We give below an SLD-refutation of $P \cup \{\leftarrow very\_much\_likes(x, y)\}$ in $KD4_s5_s$ :

| Goal | Input Clause/Rule, Mgu |
|---|---|
| $\leftarrow very\_much\_likes(x, y)$ | |
| $\leftarrow \Box_3 very\_much\_likes(x, y)$ | $\phi_{14}, \{x_1/x, y_1/y\}$ |
| $\leftarrow \Box_3 likes(x, y), \Box_3\Box_1 likes(x, y), \Box_3\Box_2 likes(x, y)$ | $\phi_{13}, \{x_2/x, y_2/y\}$ |
| $\leftarrow \Box_3\Box_1 likes(Jan, cola), \Box_3\Box_2 likes(Jan, cola)$ | $\phi_{10}, \{x/Jan, y/cola\}$ |
| $\leftarrow \Box_1 likes(Jan, cola), \Box_3\Box_2 likes(Jan, cola)$ | (7) |
| $\leftarrow \Box_3\Box_2 likes(Jan, cola)$ | $\phi_1$ |
| $\leftarrow \Box_2 likes(Jan, cola)$ | (7) |
| $\leftarrow \Box_2 likes(Jan, pepsi)$ | $\phi_8, \{x_7/Jan\}$ |
| $\diamond$ | $\phi_5$ |

## 7.2 A Schema for Semantics of $KD45_{(m)}$-MProlog

In this subsection, let $L$ denote the logic $KD45_{(m)}$. Recall that this logic is intended for reasoning about epistemic states of a number of agents. This logic is weak at the point that it has no interaction axioms between agents. On the other side, this restriction enables us to prevent expanding modal contexts of goal atoms.

In Table 7, we give a schema for semantics of $KD45_{(m)}$-MProlog. The $L$-normal form of modalities and the rules (1)–(4) and (7) in that schema are justified by the $L$-tautology $\nabla_i \phi \equiv \nabla'_i \nabla_i \phi$, where $\nabla_i$ and $\nabla'_i$ are unlabeled modal operators. The rule (6) is based on the axiom $(D)$.

**Theorem 7.2** *The schema for semantics of $KD45_{(m)}$-MProlog is correct.*

The proof of this theorem is given in Appendix D.

## 7.3 A Schema for Semantics of $KD4I_g5_a$-MProlog

In this subsection, we concern semantics of $KD4I_g5_a$-MProlog. The logic $KD4I_g5_a$ extend the logic $KD45_{(m)}$ with modal operators for groups of agents and some interaction axioms between

| $L = KD45_{(m)}$, $L$-MProlog | | |
|---|---|---|
| $\preceq_L$ is defined by Definition 4.8. | | |
| $\nabla^{(1)}_{i_1} \ldots \nabla^{(k)}_{i_k}$ is in the $L$-normal form if $i_j \neq i_{j+1}$ for all $1 \leq j < k$. | | |
| Both sides of each rule given below are in the almost $L$-normal labeled form. | | |
| $Ext_L$ | no rules | |
| $Sat_L$ | $\triangle\Box_i E \to \triangle\Box_i\Box_i E$ | (1) |
| | $\triangle\langle F\rangle_i E \to \triangle\Box_i\Diamond_i E$ | (2) |
| $NF_L$ | $\triangle\nabla_i\nabla'_i E \to \triangle\nabla'_i E$ if $\nabla'_i$ is of the form $\Box_i$ or $\langle E\rangle_i$ | (3) |
| $rNF_L$ | $\triangle\nabla_i E \leftarrow \triangle\langle X\rangle_i\nabla_i E$ if $\nabla'_i$ is of the form $\Box_i$ or $\langle E\rangle_i$ and $X$ is a fresh atom variable | (4) |
| $rSat_L$ | $\triangle\Diamond_i E \leftarrow \triangle\langle X\rangle_i E$ for $X$ being a fresh atom variable | (5) |
| | $\triangle\nabla_i\alpha \leftarrow \triangle\Box_i\alpha$ | (6) |
| | $\triangle\nabla_i\nabla'_i E \leftarrow \triangle\nabla'_i E$ if $\nabla'_i$ is of the form $\Box_i$ or $\Diamond_i$ | (7) |

Table 7: A schema for semantics of $KD45_{(m)}$-MProlog

groups and agents. Without the axiom $(5_a)$, the logic $KD4I_g5_a$ is similar to $KDI4$, except that, in $KDI4$ we have the axiom $(I) : \Box_i\phi \to \Box_j\phi$ if $i > j$, while in $KD4I_g5_a$ we have $(I_g) : \Box_i\phi \to \Box_j\phi$ if $g(i) \supset g(j)$. Thus, the logic $KD4I_g5_a$ can be viewed as a combination of $KD45_{(m)}$ and $KDI4$.

In Table 8, we give a schema for semantics of $KD4I_g5_a$-MProlog. It is a combination of the schemata for semantics of $KDI4$-MProlog and $KD45_{(m)}$-MProlog given in Tables 3 and 7, with the modifications that, instead of comparing modal indexes $i$ and $j$ we compare the sets $g(i)$ and $g(j)$, and the rules based on the axiom (5) are modified so that they are applicable only when the involved modal index stands for an agent.

When resolving a goal with an input clause, if we relax the condition that the mgu $\theta$ unifies the selected head atom $A'$ with the forward labeled form $A''$ of the head of the input clause, but only require that $\theta$ is a most general substitution such that $A'\theta$, $A''\theta$ have the same classical atom and $A'\theta$ is an $L$-instance of $A''\theta$, then we can delete the rule (8) and replace the rule (10) by $\triangle\nabla_i\nabla'_j\alpha \leftarrow \triangle\Box_k\alpha$, where $g(k) = g(i) \cup g(j)$.

Correctness of the schema for semantics of $KD4I_g5_a$-MProlog can be proved in a similar way as for the cases of $KD45_{(m)}$ and $KDI4$.

# 8 Schemata for Semantics of MProlog-$\Box$

Recall that in MProlog-$\Box$ programs and goals, existential modal operators are disallowed. In this section, we simplify the schemata given in the two previous sections for $L$-MProlog-$\Box$, where $L$ is one the considered multimodal logics about belief. Let $P$ denote an $L$-MProlog-$\Box$ program and $G$ an $L$-MProlog-$\Box$ goal.

Observe that if a schema for semantics of $L$-MProlog does not involve existential modal operators (unlabeled or labeled), then:

- It can be proved by induction on $n$ that $T_{L,P}\uparrow n$ contains only atoms of the form $\triangle E$ with $\triangle$ being a universal modality. Hence, the same claim can be said for $I_{L,P}$.
- Worlds of $M_{L,P}$ are identified by modalities of the form $\langle\top\rangle_{i_1} \ldots \langle\top\rangle_{i_k}$.

| $L = KD4I_g5_a$, $L$-MProlog |
|---|
| $\preceq_L$ is defined by Definition 4.8.<br><br>$\nabla^{(1)}_{i_1} \ldots \nabla^{(k)}_{i_k}$ is in the $L$-normal form if<br>for all $1 \leq j < k$ if $g(i_j)$ is singleton then $i_j \neq i_{j+1}$.<br><br>Both sides of each rule given below are in the almost $L$-normal labeled form.<br><br>$Ext_L$ $\triangle\Box_i\alpha \to \triangle\Box_j\alpha$ if $g(i) \supset g(j)$ (1)<br>$\triangle\Box_i\alpha \to \triangle\Box_i\Box_i\alpha$ (2)<br><br>$Sat_L$ the rules specifying $Ext_L$ plus<br>$\triangle\langle F\rangle_i E \to \triangle\Box_i\Diamond_i E$ if $g(i)$ is singleton (3)<br>$\triangle\nabla\nabla' E \to \triangle\Diamond_i E$ if $\Diamond_i \preceq_L \nabla$ and $\Diamond_i \preceq_L \nabla'$ (4)<br><br>$NF_L$ $\triangle\nabla_i\nabla'_i E \to \triangle\nabla'_i E$ if $g(i)$ is singleton and<br>$\nabla'_i$ is of the form $\Box_i$ or $\langle E\rangle_i$ (5)<br><br>$rNF_L$ $\triangle\nabla_i E \leftarrow \triangle\langle X\rangle_i\nabla_i E$ if $g(i)$ is singleton,<br>$\nabla'_i$ is of the form $\Box_i$ or $\langle E\rangle_i$, and $X$ is a fresh atom variable (6)<br><br>$rSat_L$ $\triangle\Diamond_i E \leftarrow \triangle\langle X\rangle_i E$ for $X$ being a fresh atom variable (7)<br>$\triangle\nabla_i\alpha \leftarrow \triangle\Box_j\alpha$ if $g(i) \subseteq g(j)$ (8)<br>$\triangle\Diamond_i E \leftarrow \triangle\Diamond_j E$ if $g(i) \supset g(j)$ (9)<br>$\triangle\Box_i\Box_i\alpha \leftarrow \triangle\Box_i\alpha$ (10)<br>$\triangle\nabla_i\Diamond_i E \leftarrow \triangle\Diamond_i E$ if $g(i)$ is singleton (11)<br>$\triangle\Diamond_i E \leftarrow \triangle\langle X\rangle_j\Diamond_i E$ if $g(i) \supseteq g(j)$ and<br>$X$ is a fresh atom variable (12) |

Table 8: A schema for semantics of $KD4I_g5_a$-MProlog

- Existential modal operators never appear in SLD-derivations of $P \cup \{G\}$ in $L$.

For $L$-MProlog-$\Box$, assume the word $L$-MProlog in Section 4 is replaced by $L$-MProlog-$\Box$. Define that a schema for semantics of $L$-MProlog-$\Box$ is *correct* if the main theorems 4.3, 4.9, 4.16 and 4.23 are true for $L$-MProlog-$\Box$.

**Case $L \in \{KDI4_s, KDI4\}$**

Observe that:

- The rules (3) in Tables 2 and 3 are unnecessary for computing $I_{L,P}$.
- The rules (4), (6), (8) in Tables 2 and 3 are never used in SLD-derivations of $P \cup \{G\}$ in $L$.
- Changing the rule (5) in Tables 2 and 3 to $\triangle\Box_i\alpha \leftarrow \triangle\Box_j\alpha$ if $i < j$ and the rule (7) in Table 2 to $\triangle\Box_j\Box_i\alpha \leftarrow \triangle\Box_i\alpha$ does not affect SLD-derivations of $P \cup \{G\}$ in $L$.

Hence, the schemata given in Tables 2 and 3 can be simplified respectively for $KDI4_s$-MProlog-$\Box$ and $KDI4$-MProlog-$\Box$ to the schemata given in Table 9, and these new schemata are correct.

**Case $L = KDI4_s5$**

Observe that:

| $L = KDI4_s$, $L$-MProlog-$\Box$ | |
|---|---|
| $\preceq_L$ is specified by $\Box_i \preceq_L \Box_j$ for $i \leq j$. | |
| No restrictions on the $L$-normal form of modalities. | |
| No rules specifying $NF_L$ and $rNF_L$. | |
| Rules specifying | |
| $Ext_L$ and $Sat_L$ | |
| | $\triangle\Box_i\alpha \to \triangle\Box_j\alpha$ for $i > j$ |
| | $\triangle\Box_i\alpha \to \triangle\Box_j\Box_i\alpha$ |
| $rSat_L$ | $\triangle\Box_j\alpha \leftarrow \triangle\Box_i\alpha$ for $i > j$ |
| | $\triangle\Box_j\Box_i\alpha \leftarrow \triangle\Box_i\alpha$ |

| $L = KDI4$, $L$-MProlog-$\Box$ | |
|---|---|
| $\preceq_L$ is specified by $\Box_i \preceq_L \Box_j$ for $i \leq j$. | |
| No restrictions on the $L$-normal form of modalities. | |
| No rules specifying $NF_L$ and $rNF_L$. | |
| Rules specifying | |
| $Ext_L$ and $Sat_L$ | |
| | $\triangle\Box_i\alpha \to \triangle\Box_j\alpha$ for $i > j$ |
| | $\triangle\Box_i\alpha \to \triangle\Box_i\Box_i\alpha$ |
| $rSat_L$ | $\triangle\Box_j\alpha \leftarrow \triangle\Box_i\alpha$ for $i > j$ |
| | $\triangle\Box_i\Box_i\alpha \leftarrow \triangle\Box_i\alpha$ |

Table 9: Schemata for semantics of MProlog-$\Box$, part 1

- The rule (3) in Table 4 is unnecessary for computing $I_{L,P}$.
- Changing the rule (4) in Table 4 to $\Box_i\Box_j E \to \Box_j E$ does not affect $I_{L,P}$.
- The rule (5) in Table 4 can be simulated by the rule $\Box_i E \leftarrow \Box_1\Box_i E$ so that existential modal operators never appear in SLD-derivations of $P \cup \{G\}$ in $L$. Consequently, the rules (6) and (8) in Table 4 are never used in SLD-derivations of $P \cup \{G\}$ in $L$, and the rule (9) in Table 4 can be changed to $\Box_i\Box_j E \leftarrow \Box_j E$.
- With the presence of the rule $\Box_i E \leftarrow \Box_j E$ for $i < j$, the rule $\Box_i\Box_k E \leftarrow \Box_j\Box_k E$ for $i < j$ is useless and the rule $\Box_k\Box_i E \leftarrow \Box_k\Box_j E$ for $i < j$ can be simulated. Hence the rule (7) in Table 4 can be replaced by $\Box_i E \leftarrow \Box_j E$ with $i < j$.

Hence, the schema given in Table 4 can be simplified for $KDI4_s5$-MProlog-$\Box$ to the corresponding schema given in Table 10, and this latter schema is correct.

**Case $\boldsymbol{L} = KD4_s5_s$**

Observe that:

- The rule (2) in Table 6 is unnecessary for computing $I_{L,P}$.
- Changing the rule (3) in Table 6 to $\Box_i\Box_j E \to \Box_j E$ does not affect $I_{L,P}$.
- The rule (4) in Table 6 can be simulated by the rule $\Box_i E \leftarrow \Box_j\Box_i E$ so that existential modal operators never appear in SLD-derivations of $P \cup \{G\}$ in $L$. Consequently, the rules (5) and (6) in Table 6 are never used in SLD-derivations of $P \cup \{G\}$ in $L$, and the rule (7) can be changed to $\Box_j\Box_i E \leftarrow \Box_i E$.

Hence, the schema given in Table 6 can be simplified for $KD4_s5_s$-MProlog-$\Box$ to the corresponding schema given in Table 10, and this latter schema is correct.

| $L = KDI4_s5$, $L$-MProlog-$\Box$ | |
|---|---|
| $\preceq_L$ is specified by $\Box_i \preceq_L \Box_j$ for $i \leq j$. | |
| A modality is in the $L$-normal form if its length $\leq 1$. | |
| Rules specifying | |
| $Ext_L$ | $\Box_i E \to \Box_j E$ for $i > j$ |
| $Sat_L$ | the above rule plus |
| | $\Box_i E \to \Box_m \Box_i E$ |
| $NF_L$ | $\Box_i \Box_j E \to \Box_j E$ |
| $rNF_L$ | $\Box_i E \leftarrow \Box_1 \Box_i E$ |
| $rSat_L$ | $\Box_i E \leftarrow \Box_j E$ for $i < j$ |
| | $\Box_i \Box_j E \leftarrow \Box_j E$ |

| $L = KD4_s5_s$, $L$-MProlog-$\Box$ | |
|---|---|
| $\preceq_L$ is the identity binary relation. | |
| A modality is in the $L$-normal form if its length $\leq 1$. | |
| Rules specifying | |
| $Ext_L$ | no rules |
| $Sat_L$ | $\Box_i E \to \Box_j \Box_i E$ |
| $NF_L$ | $\Box_i \Box_j E \to \Box_j E$ |
| $rNF_L$ | $\Box_j E \leftarrow \Box_i \Box_j E$ |
| $rSat_L$ | $\Box_j \Box_i E \leftarrow \Box_i E$ |

Table 10: Schemata for semantics of MProlog-$\Box$, part 2

**Case $\boldsymbol{L} = KDI45$**

Observe that:

- The rules (5), (6) and (7) in Table 5 are unnecessary for computing $I_{L,P}$.
- Changing the rule (8) in Table 5 to $\triangle\Box_i\Box_j E \to \triangle\Box_j E$ if $i \leq j$ does not affect $I_{L,P}$.
- The rule (9) in Table 5 can be simulated by the rule $\triangle\Box_j E \leftarrow \triangle\Box_i\Box_j E$ if $i \leq j$ so that existential modal operators never appear in SLD-derivations of $P \cup \{G\}$ in $L$. Consequently, the rules (10), (12), (14) and (16) in Table 5 are never used in SLD-derivations of $P \cup \{G\}$ in $L$, and the rule (11) can be changed to $\triangle\Box_i E \leftarrow \triangle\Box_j E$ if $i < j$.

Hence, the schema given in Table 5 can be simplified for $KDI45$-MProlog-$\Box$ to the schema given in Table 11, and this latter schema is correct.

**Case $\boldsymbol{L} = KD45_{(m)}$**

Observe that:

- The rule (2) in Table 7 is unnecessary for computing $I_{L,P}$.
- Changing the rule (3) in Table 7 to $\triangle\Box_i\Box_i E \to \triangle\Box_i E$ does not affect $I_{L,P}$.
- The rule (4) in Table 7 can be simulated by the rule $\triangle\Box_i E \leftarrow \triangle\Box_i\Box_i E$ so that existential modal operators never appear in SLD-derivations of $P \cup \{G\}$ in $L$. Consequently, the rules (5) and (6) in Table 7 are never used in SLD-derivations of $P \cup \{G\}$ in $L$, and the rule (7) can be changed to $\triangle\Box_i\Box_i E \leftarrow \triangle\Box_i E$.

Hence, the schema given in Table 7 can be simplified for $KD45_{(m)}$-MProlog-$\Box$ to the corresponding schema given in Table 12, and this latter schema is correct.

| $L = KDI45$, $L$-MProlog-$\Box$ | |
|---|---|
| $\preceq_L$ is specified by $\Box_i \preceq_L \Box_j$ if $i \leq j$. | |
| $\Box_{i_1} \ldots \Box_{i_k}$ is in the $L$-normal form if $i_1 > \ldots > i_k$. | |
| The following rules are accompanied with the condition that the atoms in both sides are in the $L$-normal labeled form for rules specifying $Ext_L$ and in the almost $L$-normal labeled form for other rules. (*) | |
| $Ext_L$ | $\triangle\Box_i\alpha \to \triangle\Box_j\alpha$ if $i > j$ |
| | $\triangle\Box_i\alpha \to \triangle\Box_i\Box_j\alpha$ if $i > j$ |
| | $\triangle\Box_i\Box_j\alpha \to \triangle\Box_j\alpha$ if $i > j$ |
| $Sat_L$ | the rules for $Ext_L$ with the modification stated in (*) |
| | $\triangle\Box_i E \to \triangle\Box_i\Box_i E$ |
| $NF_L$ | $\triangle\Box_i\Box_j E \to \triangle\Box_j E$ if $i \leq j$ |
| $rNF_L$ | $\triangle\Box_j E \leftarrow \triangle\Box_i\Box_j E$ if $i \leq j$ |
| $rSat_L$ | $\triangle\Box_i\alpha \leftarrow \triangle\Box_j\alpha$ if $i < j$ |
| | $\triangle\Box_i\Box_j\alpha \leftarrow \triangle\Box_i\alpha$ if $i \geq j$ |
| | $\triangle\Box_i\alpha \leftarrow \triangle\Box_j\Box_i\alpha$ if $i < j$ |

Table 11: Schemata for semantics of MProlog-$\Box$, part 3

**Case $\boldsymbol{L} = KD4I_g5_a$**

Observe that:

- The rules (3) and (4) in Table 8 is unnecessary for computing $I_{L,P}$.
- Changing the rule (5) in Table 8 to $\triangle\Box_i\Box_i E \to \triangle\Box_i E$ if $g(i)$ is singleton does not affect $I_{L,P}$.
- The rule (6) in Table 8 can be simulated by the rule $\triangle\Box_i E \leftarrow \triangle\Box_i\Box_i E$ if $g(i)$ is singleton, so that existential modal operators never appear in SLD-derivations of $P \cup \{G\}$ in $L$. Consequently, the rules (7), (9), (11) and (12) in Table 8 are never used in SLD-derivations of $P \cup \{G\}$ in $L$, and the rule (8) can be changed to $\triangle\Box_i\alpha \leftarrow \triangle\Box_j\alpha$ if $g(i) \subseteq g(j)$.

Hence, the schema given in Table 8 can be simplified for $KD4I_g5_a$-MProlog-$\Box$ to the corresponding schema given in Table 12, and this latter schema is correct.

In summary, we have the following theorem:

**Theorem 8.1** *The schemata for semantics of $L$-MProlog-$\Box$ are correct, for $L \in \{KDI4_s$, $KDI4$, $KDI4_s5$, $KDI45$, $KD4_s5_s$, $KD45_{(m)}$, $KD4I_g5_a\}$.*

# 9 MDatalog and Modal Deductive Databases

In this section, we apply the results on MProlog for deductive databases to form a query language called MDatalog and give computational methods for modal deductive databases. We show that the data complexity of MDatalog in the logics $KDI4_s5$, $KDI45$, $KD4_s5_s$, and $KD45_{(m)}$ is in PTIME. In [27], we presented MDatalog for monomodal deductive databases and gave a computing method that is based on building model graphs. The technique used in this section is at a higher level, as it is based on modal relational algebras and the magic-set transformation.

In this section, let $L$ be one of the multimodal logics considered in this work. We first define the $L$-MDatalog language. Informally, $L$-MDatalog is a sublanguage of $L$-MProlog that is free from function symbols and satisfies the condition that $L$-MDatalog program clauses are *allowed*.

**Definition 9.1** An MProlog program clause without function symbols is said to be *allowed* if every variable occurring in the head also occurs in the body. An *$L$-MDatalog program* is an $L$-MProlog

| $L = KD45_{(m)}$, $L$-MProlog-$\Box$ | |
|---|---|
| $\preceq_L$ is the identity binary relation. | |
| $\Box_{i_1} \ldots \Box_{i_k}$ is in the $L$-normal form if $i_j \neq i_{j+1}$ for all $1 \leq j < k$. | |
| Both sides of each rule given below are required to be in the almost $L$-normal labeled form. | |
| $Ext_L$ | no rules |
| $Sat_L$ | $\triangle\Box_i E \to \triangle\Box_i\Box_i E$ |
| $NF_L$ | $\triangle\Box_i\Box_i E \to \triangle\Box_i E$ |
| $rNF_L$ | $\triangle\Box_i E \leftarrow \triangle\Box_i\Box_i E$ |
| $rSat_L$ | $\triangle\Box_i\Box_i E \leftarrow \triangle\Box_i E$ |

| $L = KD4I_g5_a$, $L$-MProlog-$\Box$ | |
|---|---|
| $\preceq_L$ is defined by $\Box_i \preceq \Box_j$ if $g(i) \subseteq g(j)$. | |
| $\Box_{i_1} \ldots \Box_{i_k}$ is in the $L$-normal form if for all $1 \leq j < k$ if $g(i_j)$ is singleton then $i_j \neq i_{j+1}$. | |
| Both sides of each rule given below are required to be in the almost $L$-normal labeled form. | |
| $Ext_L$ and $Sat_L$ | |
| | $\triangle\Box_i\alpha \to \triangle\Box_j\alpha$ if $g(i) \supset g(j)$ |
| | $\triangle\Box_i\alpha \to \triangle\Box_i\Box_i\alpha$ |
| $NF_L$ | $\triangle\Box_i\Box_i E \to \triangle\Box_i E$ if $g(i)$ is singleton |
| $rNF_L$ | $\triangle\Box_i E \leftarrow \triangle\Box_i\Box_i E$ if $g(i)$ is singleton |
| $rSat_L$ | $\triangle\Box_i\alpha \leftarrow \triangle\Box_j\alpha$ if $g(i) \subseteq g(j)$ |
| | $\triangle\Box_i\Box_i\alpha \leftarrow \triangle\Box_i\alpha$ |

Table 12: Schemata for semantics of MProlog-$\Box$, part 4

program that is free from function symbols and whose clauses are allowed. An *$L$-MDatalog goal* is an $L$-MProlog goal that is free from function symbols.

We now give some definitions for modal deductive databases.

**Definition 9.2** An $n$-ary *$L$-tuple* is an ordered pair $(\triangle, \mathfrak{t})$, where $\mathfrak{t}$ is a classical $n$-ary tuple of constant symbols and $\triangle$ is a ground modality in the almost $L$-normal labeled form. An $n$-ary *$L$-relation* is a set of $n$-ary $L$-tuples. An $L$-relation is an $n$-ary $L$-relation for some $n$.

**Definition 9.3** An $L$-relation is said to be in the *$L$-normal form* if every one of its tuples is of the form $(\triangle, \mathfrak{t})$ with $\triangle$ in the $L$-normal labeled form.

A deductive database consists of two parts: extentional, and intentional. The extentional part (*edb*) is an instance of extentional relations, while the intentional part (*idb*) is a program defining intentional relations. Each predicate in a deductive database is categorized either as extentional or as intentional. A predicate $R$ is extentional w.r.t. an MDatalog program $P$ if it is not defined by $P$ (i.e. no clause of $P$ has $R$ in its head). If $R$ is defined by $P$ then it is intentional w.r.t $P$. A modal deductive database in $L$ consists of an $L$-MDatalog program $P$ and an instance $I$ of $L$-relations, which are extentional w.r.t. $P$.

Let $\mathcal{R}$ be a set of predicate symbols. An instance $I$ of $L$-relations of $\mathcal{R}$ can be treated as a set of atoms of predicates of $\mathcal{R}$. Conversely, a set $I$ of ground atoms of predicates of $\mathcal{R}$ which are in the almost $L$-normal labeled form can be treated as an instance of $L$-relations of $\mathcal{R}$. If $I$ is an instance of $L$-relations of $\mathcal{R}$ and $R$ is a predicate symbol of $\mathcal{R}$, then by $I(R)$ we denote the instance of the $L$-relation $R$ contained in $I$.

An $L$-MDatalog program $P$ can be treated as a function (denoted by $P_L$) that maps an instance of *edb* $L$-relations to an instance of *idb* $L$-relations such that $P_L(I)$ is the least (w.r.t. $\subseteq$) $L$-model generator $J$ such that $T_{L,P}(I \cup J) = J$. Let $T_{L,P,I}$ be the operator defined by $T_{L,P,I}(J) = T_{L,P}(I \cup J)$. Then $T_{L,P,I}$ is monotonic and continuous, and $P_L(I)$ is the least fixpoint of $T_{L,P,I}$ specified by $T_{L,P,I}\uparrow\omega = \bigcup_{0 \leq k \leq \omega} T_{L,P,I}\uparrow k$, where $T_{L,P,I}\uparrow k$ is defined in a similar way as $T_{L,P}\uparrow k$.

**Definition 9.4** We define an *L-MDatalog query* to be a pair $(P, \phi)$, where $P$ is an $L$-MDatalog program and $\phi$ is an $L$-MDatalog clause $\triangle(query(x_1, \ldots, x_k) \leftarrow B_1, \ldots, B_h)$ such that: *query* is a special predicate symbol not occurring in $P$ and the body of $\phi$; $x_1, \ldots, x_k$ are different variables; and $h \geq 1$. An $L$-MDatalog query $(P, \phi)$ takes as input an instance $I$ of *edb* $L$-relations and returns as output the $L$-relation $P'_L(I)(query)$, where $P' = P \cup \{\phi\}$.

## 9.1 Data Complexity of MDatalog

The *data complexity* of $L$-MDatalog is said to be in a complexity class $\mathcal{C}$ if the complexity of computing every $L$-MDatalog query is in $\mathcal{C}$. Here the query is fixed and the complexity is measured w.r.t. the size of input. Recall that the length of a formula is the number of symbols occurring in that formula. We define the size of a formula set to be the sum of the lengths of its formulas, and the size of an instance $I$ of $L$-relations to be the size of the set of atoms specified by $I$.

**Theorem 9.1** *The data complexity of L-MDatalog is in PTIME for* $L \in \{KDI4_s5,\ KDI45,\ KD4_s5_s,\ KD45_{(m)}\}$.

*Proof.* Let $(P_0, \phi)$ be an $L$-MDatalog query, and $I_0$ an input to $(P_0, \phi)$. Let $P = P_0 \cup \{\phi\}$, $c$ be the size of $P$, and $n$ the size of $P \cup I_0$. It is sufficient to show that the complexity of computing $P_L(I_0)$ is bounded by a polynomial of $n$.

Let $I = T_{L,P,I_0}\!\uparrow\! k$ for some $k \geq 1$ and let $\alpha \in I$. Then the modal depth of $\alpha$ is bounded by 1 for $L \in \{KDI4_s5, KD4_s5_s\}$, by $m$ for $L = KDI45$, and by $c$ for $L = KD45_{(m)}$[8]. Denote this bounce by $d$. The number of classical atoms that may occur in (the atoms of) $I$ is of rank $O(n^c)$. Hence the size of $I$ is of rank $O(n^{c(d+1)})$. It follows that the size of $Sat_L(I_0 \cup I)$ and the number of steps needed for computing $Sat_L(I_0 \cup I)$ from $I_0$ and $I$ are also of rank $O(n^{c(d+1)})$. The number of steps needed for computing $T_{0L,P}(Sat_L(I_0 \cup I))$ from $Sat_L(I_0 \cup I)$ is of rank $O(n^{c \cdot c \cdot (d+1)})$. The size of $T_{0L,P}(Sat_L(I_0 \cup I))$ can be estimated in a similar way as the size of $I$ and is of rank $O(n^{c(d+2)})$. The number of steps need for computing $T_{L,P,I_0}\!\uparrow\!(k+1)$ from $T_{0L,P}(Sat_L(I_0 \cup I))$ is of the same rank as the size of $T_{0L,P}(Sat_L(I_0 \cup I))$. Therefore the number of steps needed to compute $T_{L,P,I_0}\!\uparrow\!(k+1)$ from $T_{L,P,I_0}\!\uparrow\! k$ is bounded by a polynomial of $n$. The size of $T_{L,P,I_0}\!\uparrow\!\omega$ can be estimated in the same way as the size of $I$ and is of rank $O(n^{c(d+1)})$. Hence the number of steps needed to compute $T_{L,P,I_0}\!\uparrow\!\omega$ is bounded by a polynomial of $n$. □

For complexity of the Horn fragment in monomodal logics, see [10, 12, 26].

## 9.2 Modal Relational Algebras

In this subsection, we first define a modal relational algebra in $L$, called the $L$-SPCU algebra. These algebras extend the classical SPCU algebra (see, e.g., [1]) with some operators involving modalities. We then compare $L$-SPCU algebra queries with nonrecursive $L$-MDatalog programs (see Definition 9.6).

The *L-SPCU algebra* is formed by the following operators:

**Selection** The two primitive forms are $\sigma_{j=c}$ and $\sigma_{j=k}$, where $j$, $k$ are positive integers and $c$ is a constant symbol. The operator $\sigma_{j=c}$ takes as input any $L$-relation $I$ with arity $\geq j$ and returns as output an $L$-relation of the same arity. In particular, $\sigma_{j=c}(I) = \{(\triangle, \mathfrak{t}) \mid (\triangle, \mathfrak{t}) \in I \text{ and } \mathfrak{t}(j) = c\}$. The operator $\sigma_{j=k}$ is defined analogously for inputs with arity $\geq max\{j, k\}$.

**Projection** The general form of this operator is $\pi_{j_1,\ldots,j_n}$, where $j_1, \ldots, j_n$ is a sequence of positive integers, possibly with repeats. This operator takes as input any $L$-relation with arity $\geq max\{j_1, \ldots, j_n\}$ and returns an $L$-relation with arity $n$. In particular, $\pi_{j_1,\ldots,j_n}(I) = \{(\triangle, \langle c_1, \ldots, c_n\rangle) \mid (\triangle, \mathfrak{t}) \in I \text{ for some } \mathfrak{t} \text{ with } \mathfrak{t}(j_i) = c_i \text{ for } 1 \leq i \leq n\}$.

**Cross-product** This operator, denoted by $\times$, takes as input a pair of $L$-relations in the $L$-normal form having arbitrary arities $k$ and $h$ and returns an $L$-relation with arity $k+h$. In particular, if $arity(I) = k$ and $arity(J) = h$, then $I \times J = \{(\triangle, \langle \mathfrak{t}(1), \ldots, \mathfrak{t}(k), \mathfrak{s}(1), \ldots, \mathfrak{s}(h)\rangle) \mid$ there

[8] for $L = KD45_{(m)}$, it is bounded by the maximal modal depth of clauses of $P$

exist $\triangle'$ and $\triangle''$ such that $(\triangle', \mathfrak{t}) \in I$, $(\triangle'', \mathfrak{s}) \in J$, and $\triangle$ is a maximal $L$-instance in the $L$-normal labeled form of $\triangle'$ and $\triangle''\}$.

**Union** This operator, denoted by $\cup$, takes as input a pair of $L$-relations with the same arity and returns an $L$-relation with the same arity that is the union of the input relations.

**Context-shrink** The two primitive forms are $\Box_i$ and $\Diamond_i$, where $1 \leq i \leq m$. These operators take as input any $L$-relation $I$ and return as output an $L$-relation of the same arity. In particular, $\Box_i(I) = \{(\triangle, \mathfrak{t}) \mid$ there exists $(\triangle\nabla, \mathfrak{t}) \in I$ such that $\Box_i \preceq_L \nabla\}$. The operator $\Diamond_i$ is defined analogously.

**Context-stretch** The two primitive forms are $\Box_i^{\leftarrow}$ and $\Diamond_i^{\leftarrow}$, where $1 \leq i \leq m$. These operators take as input any $L$-relation $I$ in the $L$-normal form and return as output an $L$-relation of the same arity. In particular, $\Box_i^{\leftarrow}(I) = \{(\triangle\Box_i, \mathfrak{t}) \mid (\triangle, \mathfrak{t}) \in I\}$ and $\Diamond_i^{\leftarrow}(I) = \{(\triangle\Diamond_i, \mathfrak{t}) \mid (\triangle, \mathfrak{t}) \in I\}$.

**Context-selection** The general form of this operator is $\sigma_\triangle$, where $\triangle$ is a universal modality in the $L$-normal form. This operator takes as input any $L$-relation $I$ in the $L$-normal form and returns as output an $L$-relation of the same arity. In particular, $\sigma_\triangle(I) = \{(\triangle', \mathfrak{t}) \mid$ there exists $(\triangle^{(1)}, \mathfrak{t}) \in I$ and a universal modality $\triangle^{(2)}$ such that $\triangle\phi \to \triangle^{(2)}\phi$ is an $L$-tautology and $\triangle'$ is a maximal $L$-instance in the $L$-normal labeled form of $\triangle^{(1)}$ and $\triangle^{(2)}\}$.

**Saturation** This operator, denoted by $Sat_L$, takes as input any $L$-relation $I$ in the $L$-normal form and returns as output an $L$-relation of the same arity. In particular, $Sat_L(I) = \{(\triangle, \mathfrak{t}) \mid$ there exists $\triangle'$ such that $(\triangle', \mathfrak{t}) \in I$ and $\triangle E \in Sat_L(\{\triangle' E\})$ for some $E\}$, where the latter operator $Sat_L$ acts on model generators as defined in Section 4.

**Labeling** The general form of this operator is $Label_p$, where $p$ is an $n$-ary predicate symbol. This operator takes as input any $L$-relation $I$ with arity $n$ and returns as output an $L$-relation of the same arity. In particular, $Label_p(I) = \{(\triangle, \mathfrak{t}) \mid (\triangle, \mathfrak{t}) \in I$ and $\triangle$ is not of the form $\triangle'\Diamond_i\}$ $\cup$ $\{(\triangle\langle p(c_1, \ldots, c_n)\rangle_i, \langle c_1, \ldots, c_n\rangle) \mid (\triangle\Diamond_i, \langle c_1, \ldots, c_n\rangle) \in I\}$.

**Normalization** This operator, denoted by $NF_L$, takes as input any $L$-relation $I$ and returns as output an $L$-relation in the $L$-normal form and of the same arity. In particular, $NF_L(I) = \{(\triangle, \mathfrak{t}) \mid$ there exists $\triangle'$ such that $(\triangle', \mathfrak{t}) \in I$ and $\triangle E \in NF_L(\{\triangle' E\})$ for some $E\}$, where the latter operator $NF_L$ acts on model generators as defined in Section 4.

Note that the operators $\times$, $\Box_i$, $\Diamond_i$, and $\sigma_\triangle$ are dependent on the base logic $L$. However, for simplicity we do not attach the index $L$ to these operators. For $L \in \{KDI4_s5, KD4_s5_s, KDI45, KD45_{(m)}\}$, if input consists of finite $L$-relations, then the above operations can be effectively computed and they return a finite $L$-relation. For $L \in \{KDI4_s, KDI4, KD4I_g5_a\}$, we do not have this property.

**Definition 9.5** *L-SPCU (algebra) queries* are built from input $L$-relations and unary constant relations $I_L^c = \{(\triangle, \langle c\rangle) \mid \triangle$ is a universal modality in the almost $L$-normal labeled form$\}$, where $c$ is a constant symbol, using the $L$-SPCU algebra operators.

We now compare $L$-SPCU queries with queries based on $L$-MDatalog programs.

**Definition 9.6** A predicate $p$ *directly depends* on a predicate $q$ in an $L$-MDatalog program $P$ if there exists a program clause $\phi$ of $P$ containing both $p$ and $q$ such that $p$ occurs in the head of $\phi$. Define the relation "*depends*" to be the transitive closure of the relation "directly depends". An $L$-MDatalog program $P$ is *nonrecursive* if none of its predicates depends on itself.

**Proposition 9.2** *Let $L$ be one of the multimodal logics considered in this work. Then every $L$-MDatalog query $(P, \phi)$, where $P$ is a nonrecursive $L$-MDatalog program, is equivalent to an $L$-SPCU query.*

*Proof.* Since the $L$-SPCU algebra contains the union operator, it is sufficient to show that every $L$-relation $ans$ defined by a nonrecursive $L$-MDatalog program clause is equivalent to an $L$-SPCU query. For simplicity, we show this using the following representative example

$$\triangle(\Diamond_i ans(x,x,z,a) \leftarrow \Box_j R(x,b), \Diamond_k S(x,y), T(z))$$

Let $Q = \pi_1(\sigma_{1=3}(\ \Box_j(Sat_L(\sigma_{2=b}(R))) \times \Diamond_k(Sat_L(S))\ ))$. Then $ans$ is equivalent to

$$NF_L(Label_{ans}(\Diamond_i^{\leftarrow}(\sigma_{\triangle}(Q \times Q \times Sat_L(T) \times I_L^a))))$$

□

The conversion of the above proposition does not hold because the operators $Sat_L$, $\Box_i^{\leftarrow}$ and $\Diamond_i^{\leftarrow}$ may return relations which are not in the $L$-normal form.

An additional operator that is deserved for consideration is the *redundant-elimination* operator defined by $RE_L(I) = \{(\triangle, \mathfrak{t}) \in I \mid$ there is no $(\triangle', \mathfrak{t}) \in I$ such that $\triangle' \neq \triangle$ and $\triangle$ is an $L$-instance of $\triangle'\}$. We believe that this operator has a good behaviour when used in $L$-SPCU queries.

## 9.3 Evaluation of MDatalog

In this section, we extend evaluation techniques of Datalog (see, e.g., [1]) for MDatalog. We concentrate on bottom-up techniques, in particular, the seminaive evaluation and the magic-set transformation. Our method is general, however, the specific results are proved only for the logics $KDI4_s5$, $KDI45$, $KD4_s5_s$, $KD45_{(m)}$. *In this section, let $L$ be one of these logics.*

### 9.3.1 The Seminaive Evaluation

Let $P$ be an $L$-MDatalog program and $I$ an instance of $L$-relations which are extentional w.r.t. $P$. We first give a *naive* algorithm for computing $P_L(I)$. Since $P_L(I) = T_{L,P,I}\uparrow\omega$, we can obtain $P_L(I)$ by computing $T_{L,P,I}\uparrow k$ for next values of $k$ until a fixpoint $T_{L,P,I}\uparrow k = T_{L,P,I}\uparrow(k-1)$ is reached. Suppose that we have already computed $T_{L,P,I}\uparrow k$ and the content of a relation $p$ in $T_{L,P,I}\uparrow k$ is stored in $p_k$. Then to compute $T_{L,P,I}\uparrow(k+1)$ let us consider the program $P_{k+1}$ obtained from $P$ by replacing every predicate $p$ in bodies of the clauses of $P$ by $p_k$. $P_{k+1}$ is a nonrecursive MDatalog program, and hence $P_{k+1}(I)$ can be computed using the $L$-SPCU algebra operators. The results of $P_{k+1}(I)$ are then assigned to the predicates $p_{k+1}$ to start the next round (if necessary).

In the naive algorithm, a considerable amount of redundant computation is done, as $T_{L,P,I}\uparrow k \subseteq T_{L,P,I}\uparrow(k+1)$ and each round recomputes all elements of the previous round. To avoid this situation we can apply the *seminaive* evaluation technique in a similar way as for Datalog programs. Let $P'_{k+1}$, for $k \geq 1$, be the program constructed as follows: for each clause $\triangle(A \leftarrow B_1, \ldots, B_n)$ of $P$ and each $1 \leq i \leq n$, add to $P'_{k+1}$ the clause $\triangle(A \leftarrow B'_1, \ldots, B'_{i-1}, B^*_i, B''_{i+1}, \ldots, B''_n)$, where $B'_j$ (resp. $B''_j$) is obtained from $B_j$ by replacing the predicate of $B_j$, denoted by $p$, by $p_k$ (resp. $p_{k-1}$), and $B^*_i$ is obtained from $B_i$ by replacing the predicate of $B_i$, denoted by $q$, by the predicate defined by $(q_k - q_{k-1})$. The key in this evaluation is $B^*_i$, which contains only new atoms that are derived at round $k$. Then the seminaive algorithm is the modification of the naive algorithm with $P_k$ replaced by $P'_k$ for $k \geq 2$. It is straightforward to proved that the seminaive algorithm produces $T_{L,P,I}\uparrow k$ at round $k$. This means that the seminaive algorithm is correct.

### 9.3.2 The Magic-Set Transformation Technique

We now consider the *magic-set* transformation technique for MDatalog queries. In SLD-derivations, constant symbols may be push from goals to subgoals through unification, and in this way the search space is restricted. The magic-set technique simulates that kind of search restriction for bottom-up evaluation. It rewrites a given query to another equivalent one that is more effective when used with the seminaive evaluation. To illustrate this technique, we use the following program $MRSG_0$, which is a modified version of the program RSG [1] involving with modal operators:

$$\begin{array}{l}
\Box_3(rsg(x,y) \leftarrow flat(x,y)) \\
\Box_3(rsg(x,y) \leftarrow up(x,x_1), rsg(y_1,x_1), down(y_1,y)) \\
\Box_2(rsg(x,y) \leftarrow \Box_3 up(x,x_1), \Box_2 rsg(y_1,x_1), \Box_1 down(y_1,y)) \\
\Box_1(\Diamond_1 rsg(x,y) \leftarrow up(x,x_1), \Diamond_1 rsg(y_1,x_1), down(y_1,y))
\end{array}$$

and the query specified by $\phi = (query(y) \leftarrow \Diamond_2 rsg(a,y))$.

We first consider adorned versions of programs and queries. Denote $MRSG = MRSG_0 \cup \{\phi\}$. Given a goal $\leftarrow query(y)$ to $MRSG$, it is first replaced by $\leftarrow \Diamond_2 rsg(a,y)$. As the first coordinate of the goal atom is *bound* and the second coordinate is *free*, we denote the goal by $\leftarrow \Diamond_2 rsg^{bf}(a,y)$, where the superscript '$bf$' is called an *adornment*. In the next step, the operator $\Diamond_2$ in the goal atom is labeled to become $\langle X\rangle_2$, where $X$ is a fresh atom variable. Now suppose that we want to resolve the new goal with the third clause of $MRSG_0$. To make benefits from adornments, we create an adorned version of the third clause of $MRSG_0$ and resolve the current goal with it. In that adorned clause, the atom in the head should be $rsg^{bf}(x,y)$, and because $x$ is bound and *up* is an *edb* predicate, $x_1$ will be bound. Therefore that adorned clause looks like

$$\Box_2(rsg^{bf}(x,y) \leftarrow \Box_3 up(x,x_1), \Box_2 rsg^{fb}(y_1,x_1), \Box_1 down(y_1,y))$$

Note that we do not write adornments for *edb* predicates.

The relevant adorned clauses for the query $(MRSG_0, \phi)$ are as follows:

1. $query(y) \leftarrow \Diamond_2 rsg^{bf}(a,y)$
2. $\Box_3(rsg^{bf}(x,y) \leftarrow flat(x,y))$
3. $\Box_3(rsg^{bf}(x,y) \leftarrow up(x,x_1), rsg^{fb}(y_1,x_1), down(y_1,y))$
4. $\Box_2(rsg^{bf}(x,y) \leftarrow \Box_3 up(x,x_1), \Box_2 rsg^{fb}(y_1,x_1), \Box_1 down(y_1,y))$
5. $\Box_1(\Diamond_1 rsg^{bf}(x,y) \leftarrow up(x,x_1), \Diamond_1 rsg^{fb}(y_1,x_1), down(y_1,y))$
6. $\Box_3(rsg^{fb}(x,y) \leftarrow flat(x,y))$
7. $\Box_3(rsg^{fb}(x,y) \leftarrow down(y_1,y), rsg^{bf}(y_1,x_1), up(x,x_1))$
8. $\Box_2(rsg^{fb}(x,y) \leftarrow \Box_1 down(y_1,y), \Box_2 rsg^{bf}(y_1,x_1), \Box_3 up(x,x_1))$
9. $\Box_1(\Diamond_1 rsg^{fb}(x,y) \leftarrow down(y_1,y), \Diamond_1 rsg^{bf}(y_1,x_1), up(x,x_1))$

Note that in the last three clauses, the order of atoms in the bodies is changed so that the binding of $y$ in *down* can be "passed" via $y_1$ to $rsg$ and via $x_1$ to *up*. Denote the above program by $MRSG^{ad}$.

Formally, an *adornment* $\gamma$ for an $n$-ary predicate $p$ is a sequence of letters '$b$' and '$f$' with length $n$, and $p$ *adorned by* $\gamma$ is denoted by $p^\gamma$. For $A$ being of the form $\triangle p(t_1, \ldots, t_n)$, by $A^\gamma$ we denote $\triangle p^\gamma(t_1, \ldots, t_n)$. We say that a variable $x$ is *bound* in $A^\gamma$ if there exists $1 \leq j \leq n$ such that $t_j = x$ and $\gamma(j) =$ '$b$', otherwise $x$ is *free* in $A^\gamma$. Given a clause $\phi = \triangle(A \leftarrow B_1, \ldots, B_k)$ and an adornment $\gamma$ for the predicate in $A$, the *adorned version* of $\phi$ w.r.t. $\gamma$ is $\triangle(A^\gamma \leftarrow B_1^{\gamma_1}, \ldots, B_k^{\gamma_k})$, where $\gamma_i$ is specified as follows: if $B_i$ is of the form $\triangle p(t_1, \ldots, t_n)$ and $t_j$ is a constant symbol or a variable bound in $A^\gamma$ or occurring in $B_1, \ldots, B_{j-1}$ then $\gamma_i(j) =$ '$b$', else $\gamma_i(j) =$ '$f$'.

Let $(P_0, \phi)$ be an $L$-MDatalog query. Let $\psi$ be the adorned version of $\phi$ w.r.t the adornment containing only '$f$' with the modification that the head is written without adornment. Let $P = P_0 \cup \{\phi\}$ and $P^{ad}$ be the program consisting of all adorned versions of all clauses of $P_0$ plus $\psi$. We call $P^{ad}$ the *adorned program* corresponding to the query $(P_0, \phi)$.

From now we assume the modification that if $\alpha$ is of the form $\triangle \Diamond_i p^\gamma(t_1, \ldots, t_n)$ then the *forward labeled form* of $\alpha$ is $\triangle\langle p(t_1, \ldots, t_n)\rangle_i p^\gamma(t_1, \ldots, t_n)$ instead of $\triangle\langle p^\gamma(t_1, \ldots, t_n)\rangle_i p^\gamma(t_1, \ldots, t_n)$. Under this assumption, we have the following property:

**Lemma 9.3** *Let $(P_0, \phi)$ be an L-MDatalog query, $P = P_0 \cup \phi$, and $P^{ad}$ be the corresponding adorned program. Then for every* edb *instance $I$,* idb *predicate $p$ of $P$, and adornment $\gamma$ for $p$, we have that $P_L^{ad}(I)(p^\gamma) = P_L(I)(p)$. In particular, for every* edb *instance $I$, $P_L^{ad}(I)(query) = P_L(I)(query)$.*

*Proof.* It is straightforward to prove by induction on $k$ that: for every *edb* instance $I$, *idb* predicate $p$ of $P$, and adornment $\gamma$ for $p$, we have that $T_{L,P^{ad},I} \uparrow k\,(p^\gamma) = T_{L,P,I} \uparrow k\,(p)$. This proves the lemma. $\Box$

We proceed by giving a further transformation for $P^{ad}$. Consider, for example, the 9th clause of the program $MRSG^{ad}$. As the head of the clause is $\Diamond_1 rsg^{fb}(x,y)$, "input" for the body is bound by a certain relation $\Diamond_1 input\_rsg^{fb}(y)$. From $\Diamond_1 input\_rsg^{fb}(y)$ and $down(y_1,y)$, we create $sup_2^9(y,y_1)$ as a supplement for the 2nd atom in the body of the 9th clause of $MRSG^{ad}$, which is an atom of an *idb* predicate. The clause is thus transformed to the following:

(s9.1) $\Box_1(sup_2^9(y,y_1) \leftarrow \Diamond_1 input\_rsg^{fb}(y), down(y_1,y))$
(s9.2) $\Box_1(\Diamond_1 rsg^{fb}(x,y) \leftarrow sup_2^9(y,y_1), \Diamond_1 rsg^{bf}(y_1,x_1), up(x,x_1))$

Furthermore, $sup_2^9(y,y_1)$ triggers an additional search for $\Diamond_1 rsg^{bf}(y_1,x_1)$, which in turn will trigger a search for $\Box_1 rsg^{bf}(y_1,x_1)$ (due to the axiom $D$). Thus, $sup_2^9(y,y_1)$ adds additional "input" to the search for $\Box_1 rsg^{bf}(y_1,x_1)$. Hence, we also have the following clause:

(i9.1) $\Box_1(\Box_1 input\_rsg^{bf}(y_1) \leftarrow sup_2^9(y,y_1))$

As another example, for the 8th clause of $MRSG^{ad}$ we have the following:

(s8.1) $\Box_2(sup_2^8(y,y_1) \leftarrow input\_rsg^{fb}(y), \Box_1 down(y_1,y))$
(s8.2) $\Box_2(rsg^{fb}(x,y) \leftarrow sup_2^8(y,y_1), \Box_2 rsg^{bf}(y_1,x_1), \Box_3 up(x,x_1))$
(i8.1) $\Box_2(\Box_2 input\_rsg^{bf}(y_1) \leftarrow sup_2^8(y,y_1))$

For the 2nd clause of $MRSG^{ad}$ we have:

(s2.1) $\Box_3(rsg^{bf}(x,y) \leftarrow input\_rsg^{bf}(x), flat(x,y))$

Despite that we do not write adornment for *query*, its implicit adornment contains only '$f$'. Thus, the 1st clause of $MRSG^{ad}$ will be transformed to:

(s1.1) $sup_1^1 \leftarrow input\_query$
(s1.2) $query(y) \leftarrow sup_1^1, \Diamond_2 rsg^{bf}(a,y)$
(i1.1) $\Box_2 input\_rsg^{bf}(a) \leftarrow sup_1^1$
(seed) $input\_query \leftarrow$

The *seed* rule is used to trigger a search for the query. The four above rules can be simplified, however, we keep them for the sake of convenience.

The result of transformation for all clauses of $MRSG^{ad}$ forms a new program, denoted by $MRSG^m$. Because $MRSG^m$ is quite long, we do not list all of its clauses.

We now give a formal definition of the magic-set transformation. We start by giving auxiliary notations. For an atom $A$ of the form $\triangle p^{\gamma}(t_1,\ldots,t_n)$, where $|\triangle| \leq 1$ and $i_1,\ldots,i_k$ are all the indexes such that $\gamma(i_j) =$ '$b$' for $1 \leq j \leq k$ : by $input\_A$ we denote the atom $\triangle input\_p^{\gamma}(t_{i_1},\ldots,t_{i_k})$; by $input\_blf\_A$ we denote $\Box_i p^{\gamma}(t_{i_1},\ldots,t_{i_k})$ if [9] $\triangle = \Diamond_i$, and $input\_A$ otherwise. Note that we do not write adornment for *query* but it is implicitly the one that contains only '$f$'. For an adorned clause $\phi_i = \triangle(A \leftarrow B_1,\ldots,B_k)$ and $1 \leq j \leq k$, let $Sup_j^i$ be the atom of the predicate $sup_j^i$ whose arguments are the variables that occur both in $input\_A, B_1,\ldots,B_{i-1}$ and $B_i,\ldots,B_k,A$.

Let $(P_0,\phi)$ be an $L$-MDatalog query and $P^{ad}$ the corresponding adorned program. Construct $P^m$ as follows: At the beginning let $P^m$ contain only the clause $\triangle(input\_query \leftarrow)$, where $\triangle$ is the modal context of $\phi$. Then for each clause $\phi_i = \triangle(A \leftarrow B_1,\ldots,B_k)$ of $P^{ad}$ with the property that *query* depends on the predicate of $A$ :

- If no *idb* predicate occurs in $B_1,\ldots,B_k$ then add to $P^m$ the clause $\triangle(A \leftarrow input\_A, B_1,\ldots,B_k)$.
- Otherwise, let $i_1,\ldots,i_h$ be all the indexes such that $B_{i_j}$, for $1 \leq j \leq h$, is an atom of an *idb* predicate, then add to $P^m$ the following clauses:
  - $\triangle(Sup_{i_1}^i \leftarrow input\_A, B_1,\ldots,B_{i_1-1})$,
  - $\triangle(Sup_{i_j}^i \leftarrow Sup_{i_{j-1}}^i, B_{i_{j-1}},\ldots,B_{i_j-1})$ for every $1 < j \leq h$,
  - $\triangle(A \leftarrow Sup_{i_h}^i, B_{i_h},\ldots,B_k)$,
  - $\triangle(input\_blf\_B_{i_j} \leftarrow Sup_{i_j}^i)$ for every $1 \leq j \leq h$.

Among the clauses of $P^m$ there is exactly one clause defining *query*. Denote that clause by $\phi^m$. $(P^m,\phi^m)$ is the $L$-MDatalog query obtained from $(P_0,\phi)$ by the magic-set transformation.

In order to compare $(P^m,\phi^m)$ with $(P_0,\phi)$ and obtain an equivalence we need a modification for the operator $Sat_L$. The problem is that if $\triangle E \leftarrow \triangle' E$ is an instance of a rule specifying $rSat_L$

[9] $blf$ stands for "$\Box$-lifting form"

or $rNF_L$ then we should accept also $\triangle input\_E \to \triangle' \, input\_E$. For general cases we would assume the following modification: for every rule $\delta$ specifying $rSat_L$ or $rNF_L$, consider an equivalent rule in the form $\triangle E \leftarrow \triangle' E$, replace every modal operator of the form $\langle X \rangle_i$ in $\triangle'$ by $\Box_i$ and denote the result by $\triangle E \leftarrow \triangle'' E$, then extend the set of rules specifying $Sat_L$ with the rule $\delta' = (\triangle input\_E \to \triangle'' \, input\_E)$, keeping all accompanying conditions. For example, if $\delta = (\triangle \Diamond_i E \leftarrow \triangle \langle X \rangle_i E)$ then $\delta' = (\triangle \Diamond_i \, input\_E \to \triangle \Box_i \, input\_E)$, if $\delta = (\triangle \Box_i \alpha \leftarrow \triangle \Box_j \Box_i \alpha)$ for $i < j$ then $\delta' = (\triangle \Box_i \triangle' \, input\_E \to \triangle \Box_j \Box_i \triangle' \, input\_E)$ for $i < j$. For the specific logics $KDI4_s5$, $KD4_s5_s$, $KDI45$, and $KD45_{(m)}$, we just extend the primary set of rules specifying $Sat_L$ with the following rules:

- case $L \in \{KDI4_s5, KDI45\}$ : $\triangle input\_E \to \Box_m \, input\_E$ if $|\triangle| \geq 1$;
- case $L = KD4_s5_s$ : $\triangle \nabla_i \, input\_E \to \Box_i \, input\_E$;
- case $L = KD45_{(m)}$ : $\triangle \nabla_i \triangle' \, input\_E \to \triangle \Box_i \triangle' \, input\_E$ and
  $\triangle \nabla_i \nabla'_i \, input\_E \to \triangle \Box_i \, input\_E$.

The three following auxiliary lemmas can be checked in a straightforward way.

**Lemma 9.4** *Let $\alpha$ and $\beta$ be ground atoms in the L-normal labeled form. Suppose that $\alpha$ is an L-instance of $\beta$. Then every $\alpha' \in Sat_L(\{\alpha\})$ is an L-instance of some $\beta' \in Sat_L(\{\beta\})$.*

**Lemma 9.5** *Let $\alpha$ be a ground atom in the almost L-normal form. Suppose that $\beta \in NF_L(\{\alpha\})$. Then $\alpha$ is an L-instance of some atom of $Sat_L(\{\beta\})$.*

**Lemma 9.6** *Let $\triangle E$ and $\triangle' E$ be ground atoms in the L-normal labeled form. Suppose that $\triangle E$ is an L-instance of some atom of $Sat_L(\{\triangle' E\})$. Then $\triangle' input\_E$ is an L-instance of some atom of $Sat_L(\{\triangle input\_E\})$.*

We now prove that the magic-set transformation is correct.

**Lemma 9.7** *Let $(P_0, \phi)$ be an MDatalog query, $P^{ad}$ the corresponding adorned program, and $(P^m, \phi^m)$ the result of the magic-set transformation for $(P_0, \phi)$. Let $I$ be an* edb *instance and $p$ an* idb *predicate of $P^{ad}$. Then every atom $\alpha \in P_L^m(I)$ of $p$ is an L-instance of some atom from $P_L^{ad}(I)$.*

*Proof.* It is straightforward to prove this lemma by induction on the number of steps needed to derive $\alpha$ for $P_L^m(I)$, using the observation that when transforming $P^{ad}$ to $P^m$, a clause $\phi_i = \triangle(A \leftarrow B_1, \ldots, B_k)$ with $B_{i_1}, \ldots, B_{i_h}$ being atoms of *idb* predicates is broken into
$\triangle(Sup^i_{i_1} \leftarrow input\_A, B_1, \ldots, B_{i_1 - 1})$,
$\triangle(Sup^i_{i_j} \leftarrow Sup^i_{i_{j-1}}, B_{i_{j-1}}, \ldots, B_{i_j - 1})$ for every $1 < j \leq h$,
$\triangle(A \leftarrow Sup^i_{i_h}, B_{i_h}, \ldots, B_k)$,
where $input\_A$ plays the role of an additional restriction. ☐

**Lemma 9.8** *Let $(P_0, \phi)$ be an MDatalog query, $P^{ad}$ the corresponding adorned program, $(P^m, \phi^m)$ the result of the magic-set transformation for $(P_0, \phi)$, and $I$ an* edb *instance. Suppose that $\triangle E \in P_L^{ad}(I)$, $\triangle'$ is the universal modality being a $\Box$-lifting form of $\triangle$, and $\triangle' input\_E$ is an L-instance of some atom of $Sat_L(P_L^m(I))$. Then $\triangle E$ is an L-instance of some atom of $P_L^m(I)$.*

*Proof.* Let $n$ be the smallest number such that $\triangle E \in T_{L,P^{ad},I} \uparrow n$. We prove the lemma by induction on $n$. Suppose that the assertion of the lemma holds for all $n$ with smaller value. Suppose that $\triangle E \in T_{L,P^{ad},I} \uparrow n$ is created by first applying some clause $\phi_i = \triangle''(A \leftarrow B_1, \ldots, B_k)$ of $P^{ad}$ to atoms of $Sat_L(I \cup T_{L,P^{ad},I} \uparrow (n-1))$ to create $\triangle'''(A'\theta)$, where $A'$ is the forward labeled form of $A$ and $\theta$ is the involved substitution, and then normalizing $\triangle'''(A'\theta)$. Let $1 \leq i_1 < \ldots < i_h \leq k$ be all the indexes such that $B_{i_j}$ for $1 \leq j \leq h$ are atoms of *idb* predicates. Then $P^m$ contains the following clauses:

(a) $\triangle''(Sup^i_{i_1} \leftarrow input\_A, B_1, \ldots, B_{i_1-1})$,
(b) $\triangle''(Sup^i_{i_j} \leftarrow Sup^i_{i_{j-1}}, B_{i_{j-1}}, \ldots, B_{i_j-1})$ for every $1 < j \leq h$,
(c) $\triangle''(A \leftarrow Sup^i_{i_h}, B_{i_h}, \ldots, B_k)$,
(d) $\triangle''(input\_blf\_B_{i_j} \leftarrow Sup^i_{i_j})$ for every $1 \leq j \leq h$.

Since $\triangle E \in NF_L(\{\triangle'''(A'\theta)\})$, by Lemma 9.5, $\triangle'''(input\_A\theta)$ is an $L$-instance of some atom of $Sat_L(\{\triangle input\_E\})$. By Lemma 9.4, it follows that $\triangle'''(input\_A\theta)$ is an $L$-instance of some atom of $Sat_L(P^m_L(I))$. By the clause (a), this implies that $\triangle'''(Sup^i_{i_1}\theta)$ is an $L$-instance of some atom of $P^m_L(I)$. Consequently, by the clauses (d), $\triangle'''(input\_blf\_B_{i_1}\theta)$ is an $L$-instance of some $\alpha \in T_{0L,P}(P^m_L(I))$. We have $NF_L(\{\alpha\}) \subseteq P^m_L(I)$, hence by Lemma 9.5, $\alpha$ is an $L$-instance of some atom of $Sat_L(P^m_L(I))$. Hence $\triangle'''(input\_blf\_B_{i_1}\theta)$ is an $L$-instance of some atom of $Sat_L(P^m_L(I))$.

We have that $\triangle'''(B_{i_1}\theta)$ is an $L$-instance of some atom of $Sat_L(T_{L,P^{ad},I}\uparrow(n-1))$. Let $\triangle^* B^*_{i_1} \in T_{L,P^{ad},I}\uparrow(n-1)$ be the atom such that $\triangle'''(B_{i_1}\theta)$ is an $L$-instance of some atom of $Sat_L(\{\triangle^* B^*_{i_1}\})$. Let $\triangle^{(4)} E_{i_1} = \triangle^* B^*_{i_1}$ and let $\triangle^{(5)}$ be the universal modality being a $\Box$-lifting form of $\triangle^{(4)}$. By Lemma 9.4, $\triangle'''(B_{i_1}\theta)$ is an $L$-instance of some atom of $Sat_L(\{\triangle^{(5)} E_{i_1}\})$. By Lemma 9.6, it follows that $\triangle^{(5)} input\_E_{i_1}$ is an $L$-instance of some atom of $Sat_L(\{\triangle'''(input\_B_{i_1}\theta)\})$. Hence, by Lemma 9.4, $\triangle^{(5)} input\_E_{i_1}$ is an $L$-instance of some atom of $Sat_L(\{\triangle'''(input\_blf\_B_{i_1}\theta)\})$, and then is also an $L$-instance of some atom of $Sat_L(P^m_L(I))$. Hence, by the inductive assumption, $\triangle^* B^*_{i_1}$ is an $L$-instance of some atom of $P^m_L(I)$. By Lemma 9.4, it follows that $\triangle'''(B_{i_1}\theta)$ is an $L$-instance of some atom of $Sat_L(P^m_L(I))$.

Analogously, it can be shown that, for $1 < j \leq h$, $\triangle'''(Sup^i_{i_j}\theta)$ is an $L$-instance of some atom of $P^m_L(I)$ and $\triangle'''(B_{i_j}\theta)$ is an $L$-instance of some atom of $Sat_L(P^m_L(I))$. Hence, by the clause (c), $\triangle'''(A'\theta)$ is an $L$-instance of some $\triangle^{(6)}(A'\theta) \in T_{0L,P^m}(Sat_L(I \cup P^m_L(I)))$. It can be shown that $\triangle E \in NF_L(\{\triangle'''(A'\theta)\})$ is an $L$-instance of some atom of $NF_L(\{\triangle^{(6)}(A'\theta)\})$. This is trivial for the case $L \in \{KDI4_s5, KD4_s5_s, KD45_{(m)}\}$. For the case $L = KDI45$, the claim holds due to the rule (1) of the schema given in Table 5. Therefore $\triangle E$ is an $L$-instance of some atom of $P^m_L(I)$. $\Box$

**Corollary 9.9** *Let $(P_0, \phi)$ be an MDatalog query, $P^{ad}$ the corresponding adorned program, $(P^m, \phi^m)$ the result of the magic-set transformation for $(P_0, \phi)$, and $I$ an* edb *instance. Then every atom $\triangle query(c_1, \ldots, c_k) \in P^{ad}_L(I)$ is an $L$-instance of some atom of $P^m_L(I)$.*

*Proof.* Suppose that $\triangle query(c_1, \ldots, c_k) \in P^{ad}_L(I)$. It follows that $\triangle input\_query$ is an $L$-instance of some atom of $P^m_L(I)$. Let $\triangle'$ be the universal modality that is a $\Box$-lifting form of $\triangle$. We derive that $\triangle' input\_query$ is an $L$-instance of some atom of $Sat_L(P^m_L(I))$. Hence, by the above lemma, $\triangle query(c_1, \ldots, c_k)$ is an $L$-instance of some atom of $P^m_L(I)$. $\Box$

We give below the main theorem of this subsection. It informally states that the magic-set transformation for $L$-MDatalog is correct.

**Theorem 9.10** *Let $(P_0, \phi)$ be an MDatalog query, $P = P_0 \cup \{\phi\}$, $(P^m, \phi^m)$ be the result of the magic-set transformation for $(P_0, \phi)$, and $I$ an* edb *instance. Then every atom $\triangle query(c_1, \ldots, c_k) \in P^m_L(I)$ is an $L$-instance of some atom of $P_L(I)$, and every atom $\triangle query(c_1, \ldots, c_k) \in P_L(I)$ is an $L$-instance of some atom of $P^m_L(I)$. This means that $RE_L(P^m_L(I)(query)) = RE_L(P_L(I)(query))$.*

*Proof.* This theorem immediately follows from Lemmas 9.3, 9.7, and Corollary 9.9. $\Box$

Given an $L$-MDatalog query $(P_0, \phi)$, to evaluate it we can first transform it into $(P^m, \phi^m)$ using the magic-set transformation, and then apply the seminaive evaluation for the new query.

# 10 Discussion and Conclusion

Our most important contribution in this work is the proposed framework for developing fixpoint semantics, the least model semantics, and SLD-resolution calculi for multimodal logic programs. The framework is formulated in a direct way (not using translation to the classical logic) and closely to the style of classical logic programming.

In comparison with other works that also use the direct approach for defining declarative and procedural semantics for modal logic programs, e.g. [4, 7], our work [28] and this are the first ones that do not require restrictions on occurences of modal operators. In [4] Balbiani et al gave a declarative semantics and an SLD-resolution for a class of logic programs in the monomodal logics $KD$, $T$ and $S4$. To modal programs the authors associate a declarative semantics represented by a tree which is defined as the limit of a certain transformation on modal programs. The fixpoint represents a minimal Kripke model of the program. The work requires that the $\Box$ operator does not occur in bodies of program clauses and goals. In the definition of the minimal Kripke model of a program [4], the technique of connecting each newly created world to an empty world at the time of its creation (or a similar one) is not used, hence although the minimal Kripke model of a program defined in [4] is minimal with respect to the restricted class of goals, in general it is not a least Kripke model of the program in the considered logic. There is a common point between [4] and our work: in both of the works, labeled modal operators are used to convert $\langle t\rangle(\phi \wedge \psi)$ to $\langle t\rangle\phi \wedge \langle t\rangle\psi$. Labeled modal operators in [4] come from Skolemization, and terms are used to label the $\Diamond$ operator. In our work, the labeling technique results from the technique of building model graphs, and we feel convenient to use classical atoms and atom variables to label $\Diamond_i$ operators. As mentioned in the Introduction, Baldoni et al [7] gave a framework for developing declarative and operational semantics for logic programs in multimodal logics which have axioms of the form $[t_1]\ldots[t_n]\phi \to [s_1]\ldots[s_m]\phi$, where $[t_i]$ and $[s_j]$ are universal modal operators. In that work, existential modal operators are disallowed in programs and goals. To represent worlds in canonical models of programs, the authors used sequences of universal modal operators, which are similar to sequences of $\langle\top\rangle_i$ in our work.

Debart et al [14] and Akama [2] used the functional translation method to translate (multi)modal logic programs to classical logic programs. The method does not require the axiomatization of the accessibility relations associated with modal operators, rather it directly encodes the accessibility relations into the unification algorithm. In [29] Nonnengart proposed an approach based on relational and functional translation (which is also called the semifunctional translation). The approach uses accessibility relations for translated programs and does not modify unification. These translation approaches may speed up searching answers for queries. However, in our opinion, a procedure for searching answers in logic programming should not be just a deduction system in the normal sense. Intuition of atoms occurring in derivations is also important. It is needed for the debugging and iterative modes of programming. Let us consider, for example, translation of the goals $G_1 = \leftarrow \Box p$ and $G_2 = \leftarrow \Box\Diamond p(x)$. Using any of the mentioned translation methods, $G_1$ is translated to $\leftarrow p(\tau : a)$. The goal $G_2$ is translated to $\leftarrow p(\tau : f(x) : y, x)$ using the functional translation, and to $\leftarrow p(y, x), R(\tau : f(x), y)$ using the semifunctional translation. In our opinion, the translated goals are much less intuitive than the original ones. Furthermore, if we want to let programmers to have some control in using properties of the base logic, then rules used in our approach (e.g. in the form $\triangle\Box_j\Diamond_k\alpha \leftarrow \triangle\Diamond_i\alpha$ or $\triangle\Box_i\alpha \leftarrow \triangle\Diamond_j\Box_k\alpha$) are more intuitive for them than rules used in the semifunctional translation approach (e.g. in the form $R_k(x, y) \leftarrow R_j(z, x), R_i(z, y)$). In comparison with the functional translation, the semifunctional translation has the good property that it is straightforward to convert the least Herbrand model of a translated program to the least Kripke model of the original program. It seems hard to develop the least Kripke model semantics for modal logic programs using the functional translation approach.

One of good and important features of our framework is $L$-normal form of modalities. In symmetric (multi)modal logics, e.g. $KDI4_s5$, $KDI45$, $KD4_s5_s$, $KD45_{(m)}$, it is a tool that allows us to restrict lengths of modalities appearing in derivations. Such a tool was not introduced in [4, 7, 2, 14, 29]. Due to the $L$-normal form of modalities, where $L$ is one of the four mentioned logics, we have been able to show that the intentional relations of a modal deductive database in $L$ can be computed in PTIME and have polynomial size (in the size of the extentional relations).

In comparison with our previous work [28], in this work the operators $Ext_L$, $Sat_L$, $NF_L$, $rSat_L$, and $rNF_L$ are all specified by sets of rules. This way is more declarative and better reflects axioms of the base logic. The $\Box$-lifting and backward labeling operators introduced in [28] are classified in this work as rules for specifying $rSat_L$. The definitions of $L$-instance of an atom and $L$-instance of a program clause have been also abstracted.

In literature of computer science, multimodal logics are much more studied for reasoning about knowledge than about belief (see, e.g., Fagin et al [15], Meyer and van der Hoek [23]). In this work, we have concentrated on multimodal logics intended for reasoning about belief, in particular, for reasoning about multi-degree belief, for use in distributed systems of belief, and for reasoning about epistemic states of agents in multi-agent systems. The logics about multi-degree belief proposed by us are somehow similar to graded modal logics but different at the aspect that degrees in the former case are symbolic, while grades in the latter case are numeric[10]. We think that our schemata for semantics of MProlog in the considered multimodal logics about belief are practically useful. On the other side, our schema for semantics of MProlog in the class $BSMM$ of basic serial multimodal logics is interesting from the theoretical point of view. It shows that declarative and procedural semantics of multimodal logic programs can be formulated in a direct way, not using translation to the classical logic. These schemata are another one of our main contributions.

The field of temporal deductive databases has received a lot of attentions from researchers (see, e.g., [8, 13, 19, 20, 32]), while the term "modal deductive databases" is hard to find in the literature of computer science. Our work [27] and this one are pioneers in the field. In this work, we have developed modal relational algebras and the magic-set transformation for modal deductive databases. Though relational algebras and the magic-set transformation technique are well known for classical (and even temporal) deductive databases, their extensions for modal deductive databases are not trivial. Our modal relational algebras are defined basing on model generators and the direct consequences operator of our framework. Note that for modal deductive databases, the translation approaches do not work due to Skolemization function symbols. Belief is a kind of uncertainty, hence modal deductive databases using multi-degree belief have potential use for practical applications. Distributed deductive databases can also use multimodal logics to model agents, belief, and possibility.

The most known implemented system for modal logic programming is Molog, proposed by Fariñas del Cerro and constructed by his group. The Molog language is similar to and as expressive as our eMProlog language. With Molog, the user can fix a modal logic and define or choose the rules to deal with modal operators. Molog can be viewed as a framework which can be instantiated with particular modal logics. As an extension of Molog, the *Toulouse Inference Machine* (TIM) [5] (together with an abstract machine model called TARSKI for implementation [6]) makes it possible for a user to select clauses which cannot exactly unify with the current goal, but just resemble it in some way. We have been also implementing our MProlog system. It will be available soon. Here are some main features of the system:

- MProlog is implemented in Prolog and as a module for Prolog. This means that we can mix MProlog with Prolog and use most features and libraries of Prolog. This is a big difference in comparison with Molog.
- SLD-resolution calculi can be specified using our proposed framework.
- A program may use different calculi.
- There are three kinds of predicates: classical, modal, and classical but defined using modal formulas.
- Modalities are represented as lists. For example, $\Box_1 \Box_i \Diamond_j \langle X \rangle_3 p(a)$ may be represented as $[b(1), b(I), d(J), ld(3, X)]@p(a)$, and $\Box_x god\_exists \leftarrow christian(x)$ as $[believes(X)]@god\_exists \leftarrow christian(X)$.
- Rules may be given in the form $\alpha \leftarrow \phi, \beta, \psi$, where $\phi$ is a pre-condition to be checked and $\psi$ is a post-computation.
- A resolving cycle is a derivation using a sequence of $rSat_L/rNF_L$ rules and a program clause. Shorter sequences of rules are tried before longer ones. Programmers have access to the history of the current resolving cycle.
- There are a number of options that can be used to restrict the search space.

[10]Grades are used to indicate the number of worlds accessible from the current world.

## Acknowledgements

The author would like to express his thanks to professor Andrzej Szałas for reading the draft of this paper and giving helpful comments.

## References

[1] Abiteboul, S., Hull, R., Vianu, V.: *Foundations of Databases*, Addison Wesley, 1995.

[2] Akama, S.: A Meta-Logical Foundation of Modal Logic Programming. 1-20-1, Higashi-Yurigaoka, Asao-ku, Kawasaki-shi, 215, Japan, December 1989.

[3] Apt, K.: Logic Programming, in: *Handbook of Theoretical Computer Science, Volume B: Formal Models and Semantics* (J. van Leeuwen, Ed.), Elsevier, 1990.

[4] Balbiani, P., Fariñas del Cerro, L., Herzig, A.: Declarative Semantics for Modal Logic Programs, *Proceedings of the 1988 International Conference on Fifth Generation Computer Systems*, ICOT, 1988.

[5] Balbiani, P., Herzig, A., Lima-Marques, M.: TIM: The Toulouse Inference Machine for Non-Classical Logic Programming, *PDK'91: International Workshop on Processing Declarative Knowledge*, Springer-Verlag, 1991.

[6] Balbiani, P., Herzig, A., Lima-Marques, M.: Implementing Prolog Extensions: A Parallel Inference Machine, *Proceedings of the 1992 International Conference on Fifth Generation Computer Systems*, ICOT, 1992.

[7] Baldoni, M., Giordano, L., Martelli, A.: A Framework for a Modal Logic Programming, *Joint International Conference and Symposium on Logic Programming*, MIT Press, 1996.

[8] Baudinet, M., Chomicki, J., Wolper, P.: Temporal Deductive Databases, in: *Temporal Databases: Theory, Design, and Implementation, A. Tansel et al (editors)*, Benjamin Cummings, 1993, 294–320.

[9] Blackburn, P., de Rijke, M., Venema, Y.: *Modal Logic*, Cambridge University Press, 2002.

[10] Fariñas del Cerro, L., Penttonen, M.: A Note on the Complexity of the Satisfiability of Modal Horn Clauses, *Logic Programming*, **4**, 1987, 1–10.

[11] Chellas, B.: *Modal Logic: An Introduction*, Cambridge Univ. Press, 1980.

[12] Chen, C., Lin, I.: The Computational Complexity of the Satisfiability of Modal Horn Clauses for Modal Propositional Logics, *Theoretical Computer Science*, **129**, 1994, 95–121.

[13] Chomicki, J.: Temporal Query Languages: A Survey, *Temporal Logic: ICTL'94* (D. Gabbay, H. Ohlbach, Eds.), 827, Springer-Verlag, 1994.

[14] Debart, F., Enjalbert, P., Lescot, M.: Multimodal Logic Programming Using Equational and Order-Sorted Logic, *Theoretical Computer Science*, **105**, 1992, 141–166.

[15] Fagin, R., Halpern, J. Y., Moses, Y., Vardi, M. Y.: *Reasoning About Knowledge*, MIT Press, 1995.

[16] Fisher, M., Owens, R.: An Introduction to Executable Modal and Temporal Logics, *Executable Modal and Temporal Logics, IJCAI'93 workshop, M. Fisher and R. Owens (eds.)*, Springer, 1995.

[17] Fitting, M., Mendelsohn, R. L.: *First-Order Modal Logic*, Kluwer Academic Publishers, 1999.

[18] Garson, J.: Quantification in Modal Logic, in: *Handbook of Philosophical Logic, Volume II* (F. Guenthner, D. Gabbay, Eds.), 1999, 249–307.

[19] Kanellakis, P. C., Goldin, D. Q.: Constraint Programming and Database Query Languages, *Theoretical Aspects of Computer Software*, 1994.

[20] Koubarakis, M.: *Foundations of Temporal Constraint Databases*, Ph.D. Thesis, Technical University of Crete, 1993.

[21] Leitsch, A.: *The Resolution Calculus*, Springer, 1997.

[22] Lloyd, J.: *Foundations of Logic Programming, Second Edition*, Springer-Verlag, 1987.

[23] Meyer, J.-J., van der Hoek, W.: *Epistemic Logic for Computer Science and Artificial Intelligence*, Cambridge University Press, 1995.

[24] Minker, J.: Logic and Databases: A 20-Year Retrospective, *International Workshop LID'96*, Verlag Springer, 1996.

[25] Mints, G.: Gentzen-type Systems and Resolution Rules, *P.Martin-Löf, G. Mints (eds.): COLOG-88, LNCS 417*, Springer, 1988.

[26] Nguyen, L. A.: Constructing the Least Models for Positive Modal Logic Programs, *Fundamenta Informaticae*, **42**(1), 2000, 29–60.

[27] Nguyen, L. A.: The Modal Query Language MDatalog, *Fundamenta Informaticae*, **46**(4), 2001, 315–342.

[28] Nguyen, L. A.: A Fixpoint Semantics and an SLD-Resolution Calculus for Modal Logic Programs, *Fundamenta Informaticae*, **55**(1), 2003, 63–100.

[29] Nonnengart, A.: How to Use Modalities and Sorts in Prolog, *Logics in Artificial Intelligence, European Workshop, JELIA '94, York, UK, September 5-8, 1994, Proceedings* (C. MacNish, D. Pearce, L. M. Pereira, Eds.), 838, Springer, 1994.

[30] Ohlbach, H.: A Resolution Calculus for Modal Logics, *Proceedings of CADE-88, LNCS310*, Springer, 1988.

[31] Orgun, M., Ma, W.: An Overview of Temporal and Modal Logic Programming, *D.M. Gabbay and H.J. Ohlbach, editors, Proc. First Int. Conf. on Temporal Logic - LNAI 827*, Springer-Verlag, 1994.

[32] Revesz, P.: A Closed Form for Datalog Queries with Integer (Gap)-Order Constraints, *Theoretical Computer Science*, **116**(1), 1993, 117–149.

# Appendix A: Additional Proofs

**Proof of Proposition 3.1**

For any formula $\phi$ with $n$ variables and no quantifiers, by $p_\phi$ we denote a fresh $n$-ary predicate symbol, and by $E_\phi$ we denote the classical atom $p_\phi(x_1,\ldots,x_n)$, where $x_1,\ldots,x_n$ are the variables occurring in $\phi$. In this proof we use $\Gamma$ and $\Lambda$ to denote sequences of formulas.

Let $P$ be an eMProlog program and $G = \leftarrow \xi$ an eMProlog goal. Let $P_2 = P \cup \{E_\xi \leftarrow \xi\}$ and $G' = \leftarrow E_\xi$. We first apply translation rules that replace a complicated formula by a fresh atom which is defined by that formula. The method was first used by Mints [25] for modal logics. The translation rules are given in the below list, in which each row contains a formula to be replaced in the left, and the replacing ones in the right. The symbol $\triangle$ in the rules presents a universal modality. The formula $\phi$ in the rules (1), (3) and (4), and $\psi$ in the rules (7) and (8) are required not to be a classical atom. The rules (5) and (9) are designed to shorten the result. We apply the translation rules to $P_2$ until no further changes can be made. Let $P_3$ be the resulting set.

$$\begin{array}{rll}
\forall(\triangle\Diamond_i\phi) & \forall(\triangle\Diamond_i E_\phi),\ \forall(\triangle\Box_i(\phi \leftarrow E_\phi)) & (1)\\
\forall(\triangle(\phi\wedge\psi)) & \forall(\triangle\phi),\ \forall(\triangle\psi) & (2)\\
\forall(\triangle(\Box_i\phi \leftarrow \Gamma)) & \forall(\triangle(\Box_i E_\phi \leftarrow \Gamma)),\ \forall(\triangle\Box_i(\phi \leftarrow E_\phi)) & (3)\\
\forall(\triangle(\Diamond_i\phi \leftarrow \Gamma)) & \forall(\triangle(\Diamond_i E_\phi \leftarrow \Gamma)),\ \forall(\triangle\Box_i(\phi \leftarrow E_\phi)) & (4)\\
\forall(\triangle(\phi\wedge\psi \leftarrow \Gamma)) & \forall(\triangle(E_{\phi\wedge\psi} \leftarrow \Gamma)),\ \forall(\triangle(\phi \leftarrow E_{\phi\wedge\psi})),\ \forall(\triangle(\psi \leftarrow E_{\phi\wedge\psi})) & (5)\\
\forall(\triangle((\phi \leftarrow \psi) \leftarrow \Gamma)) & \forall(\triangle(\phi \leftarrow \psi,\Gamma)) & (6)\\
\forall(\triangle(\phi \leftarrow \Gamma,\Box_i\psi,\Lambda)) & \forall(\triangle(\phi \leftarrow \Gamma,\Box_i E_\psi,\Lambda)),\ \forall(\triangle\Box_i(E_\psi \leftarrow \psi)) & (7)\\
\forall(\triangle(\phi \leftarrow \Gamma,\Diamond_i\psi,\Lambda)) & \forall(\triangle(\phi \leftarrow \Gamma,\Diamond_i E_\psi,\Lambda)),\ \forall(\triangle\Box_i(E_\psi \leftarrow \psi)) & (8)\\
\forall(\triangle(\phi \leftarrow \Gamma,\psi\vee\zeta,\Lambda)) & \forall(\triangle(\phi \leftarrow \Gamma,E_{\psi\vee\zeta},\Lambda)),\forall(\triangle(E_{\psi\vee\zeta} \leftarrow \psi)),\forall(\triangle(E_{\psi\vee\zeta} \leftarrow \zeta)) & (9)\\
\forall(\triangle(\phi \leftarrow \Gamma,\psi\wedge\zeta,\Lambda)) & \forall(\triangle(\phi \leftarrow \Gamma,\psi,\zeta,\Lambda)) & (10)
\end{array}$$

It is easily seen that $P_3$ is an MProlog program. If $L$ is not one of the considered multimodal logics of belief, then just take $P' = P_3$.

If $L$ is $KDI4_s5$ or $KD4_s5_s$, then replace every modal context $\triangle\Box_i$ in $P_3$ by $\Box_i$ (note that $\triangle\Box_i\phi \equiv \Box_i\phi$ is $L$-valid), and let $P'$ be the resulting program.

If $L = KD45_{(m)}$, then repeatedly replace every sequence $\Box_i\Box_i$ in modal contexts of program clauses of $P_3$ by $\Box_i$ (note that $\Box_i\Box_i\phi \equiv \Box_i\phi$ is $KD45_{(m)}$-valid), and let $P'$ be the resulting program. If $L = KD4I_g5_a$ then let $P'$ be specified similarly but with the condition that $g(i)$ is singleton.

For $L \in \{KDI4_s, KDI4, KDI45\}$ : Repeatedly replace every clause $\triangle\Box_i(A \leftarrow B_1,\ldots,B_k)$ in $P_3$, where $\triangle$ is not empty, by $\triangle E_{\Box_i\phi}$, $\Box_m(\Box_i E_\phi \leftarrow E_{\Box_i\phi})$, and $\Box_m(A \leftarrow B_1,\ldots,B_k,E_\phi)$, where $\phi = (A \leftarrow B_1,\ldots,B_k)$. (Note that by the axioms (4) and $(I)$, we have $\Box_m\psi \to \triangle\psi$.) Denote the resulting set by $P'$.

It is clear that for $L$ being one of the considered multimodal logics of belief, $P'$ is an $L$-MProlog program. The whole translation is done in polynomial time.

Suppose that $\theta$ is a correct answer in $L$ for $P' \cup \{G'\}$. Let $M$ be an arbitrary $L$-model of $P$. We show that $M \vDash \forall(\xi\theta)$. Let $M'$ be an $L$-model with the same frame as $M$ such that the content of each world $w$ in $M'$ is the least extension of the content of $w$ in $M$ with the property that for any formula $\phi$ with $E_\phi$ occurring in $P'$, $M', w \vDash \forall(E_\phi \leftarrow \phi)$. Thus $M'$ is an $L$-model of $P'$ and $M' \vDash \forall(E_\xi \to \xi)$. Since $\theta$ is a correct answer in $L$ for $P' \cup \{G'\}$, we have $M' \vDash \forall(E_\xi\theta)$, and hence $M' \vDash \forall(\xi\theta)$. It follows that $M \vDash \forall(\xi\theta)$ (because $M'$ differs from $M$ only on the semantics of new predicate symbols). Therefore every correct answer in $L$ for $P' \cup \{G'\}$ is a correct answer in $L$ for $P \cup \{G\}$.

For the conversion, suppose that $\theta$ is a correct answer in $L$ for $P \cup \{G\}$. Let $M'$ be an $L$-model of $P'$. Thus, $M'$ is an $L$-model of $P$ and $M' \vDash \forall(E_\xi \leftarrow \xi)$. Since $\theta$ is a correct answer in $L$ for $P \cup \{G\}$, we have $M' \vDash \forall(\xi\theta)$, and hence $M' \vDash \forall(E_\xi\theta)$. Therefore every correct answer in $L$ for $P \cup \{G\}$ is a correct answer in $L$ for $P' \cup \{G'\}$.

**Proof of the mgu lemma 4.17**

Let the unrestricted refutation of $P \cup \{G\}$ consist of a sequence $G_0 = G, G_1, \ldots, G_n$ of goals, a sequence $\phi_1, \ldots, \phi_n$ of variants of program clauses of $P$, $rSat_L$ rules, or $rNF_L$ rules, and a sequence $\theta_1, \ldots, \theta_n$ of unifiers. Let $\theta$ be the computed answer from the unrestricted refutation. We prove the result by induction on $n$.

Suppose that $n = 1$. This means that $G = \leftarrow \triangle' A'$ and the empty clause is an $L$-resolvent of $G$ and the input clause $\phi_1 = \triangle(A \leftarrow)$, where $A'$ is the selected head atom. Let $\theta'_1$ be an mgu of $A'$ and the forward labeled form of $A$. Then $\theta_1 = \theta'_1\gamma$ for some $\gamma$. Furthermore, $P \cup \{G\}$ has a refutation in $L$ consisting of $G_0 = G$, $G_1 = \diamond$ with input clause $\phi_1$ and mgu $\theta'_1$.

Now suppose that the result holds for unrestricted refutations with length less than $n$. Let $G = \leftarrow \alpha_1, \ldots, \alpha_k$ and $\alpha_i$ be the selected atom of $G$.

Suppose that $G_1$ is derived from $G$ and the input clause $\phi_1 = \triangle(A \leftarrow B_1, \ldots, B_l)$ in $L$, the selected atom is $\alpha_i = \triangle' A'$, where $A'$ is the selected head atom. There exists an mgu $\theta'_1$ for $A'$ and the forward labeled form of $A$. We have $\theta_1 = \theta'_1\delta$ for some $\delta$. Let $G'_1$ be the goal derived in the same way as $G_1$ but with $\theta'_1$ instead of $\theta_1$. We have $G_1 = G'_1\delta$. Then $G_2$ can be derived from $G'_1$ in the same way as from $G_1$ but with unifier $\delta\theta_2$ instead of $\theta_2$. Thus $P \cup \{G\}$ has an unrestricted refutation in $L$ consisting of $G_0 = G, G'_1, G_2, \ldots, G_n$ with unifiers $\theta'_1, \delta\theta_2, \theta_3, \ldots, \theta_n$. By the inductive assumption, $P \cup \{G'_1\}$ has a refutation in $L$ with mgu's $\theta'_2, \ldots, \theta'_n$ such that $\delta\theta_2 \ldots \theta_n = \theta'_2 \ldots \theta'_n\gamma$, for some $\gamma$. Thus $P \cup \{G\}$ has a refutation in $L$ consisting of $G_0 = G, G'_1, \ldots, G'_n = \diamond$ with mgu's $\theta'_1, \theta'_2 \ldots \theta'_n$ such that $\theta_1\theta_2 \ldots \theta_n = \theta'_1\delta\theta_2 \ldots \theta_n = \theta'_1\theta'_2 \ldots \theta'_n\gamma$.

The cases when $G_1$ is derived from $G$ and an $rSat_L/rNF_L$ rule variant are similar to the above case.

**Proof of the lifting lemma 4.18**

Let the refutation of $P \cup \{G\theta\}$ consist of a sequence $G_0 = G, G_1, \ldots, G_n$ of goals, a sequence $\phi_1, \ldots, \phi_n$ of variants of program clauses of $P$, $rSat_L$ rules, or $rNF_L$ rules, and a sequence $\theta_1, \ldots, \theta_n$ of mgu's.

Suppose that $G_1$ is an $L$-resolvent of $G\theta$ and the input clause $\phi_1$ using $\theta_1$. We may assume that $\theta$ does not act on any variables of $\phi_1$. Let $\phi_1 = \triangle(A \leftarrow B_1, \ldots, B_l)$, $G = \leftarrow \alpha_1, \ldots, \alpha_k$, and the selected atom of $G\theta$ be $\alpha_i\theta = (\triangle' A')\theta$, where $A'\theta$ is the selected head atom. Now $\theta\theta_1$ is a unifier for $A'$ and the forward labeled form of $A$. The result of resolving $G$ and $\phi_1$ using $\theta\theta_1$ is exactly $G_1$. Thus we obtain an unrestricted refutation of $P \cup \{G\}$ in $L$, which looks exactly like the given refutation of $P \cup \{G\theta\}$, except the original goal is different and the first unifier is $\theta\theta_1$. Now apply the mgu lemma.

The cases when $G_1$ is derived from $G$ and an $rSat_L/rNF_L$ rule are similar to the above case.

**Proof of the lemma 4.22**

Suppose $\alpha$ has variables $x_1, \ldots, x_n$. Let $a_1, \ldots, a_n$ be distinct constants not appearing in $P$ and $\alpha$, and let $\theta$ be the substitution $\{x_1/a_1, \ldots, x_n/a_n\}$. Then it is clear that $\alpha\theta$ is a logical consequence in $L$ of $P$. Since $\alpha\theta$ is ground, by Lemma 4.21, $P \cup \{\leftarrow \alpha\theta\}$ has a refutation in $L$. Since the $a_i$ do not appear in $P$ or $\alpha$, by replacing $a_i$ by $x_i$ (for $1 \leq i \leq n$) in this refutation, we obtain a refutation of $P \cup \{\leftarrow \alpha\}$ in $L$ with the identity substitution as the computed answer.

**Proof of the completeness theorem 4.23**

Suppose $G$ is the goal $\leftarrow \alpha_1, \ldots, \alpha_k$. Since $\theta$ is correct, $\forall((\alpha_1 \wedge \ldots \wedge \alpha_k)\theta)$ is a logical consequence of $P$ in $L$. By Lemma 4.22, there exists a refutation of $P \cup \{\leftarrow \alpha_i\theta\}$ in $L$ such that the computed answer is the identity substitution, for $1 \leq i \leq k$. We can combine these refutations into a refutation of $P \cup \{G\theta\}$ such that the computed answer is the identity substitution. Applying the lifting lemma, we conclude that there exists a refutation of $P \cup \{G\}$ in $L$ with computed answer $\gamma$ such that $\theta = \gamma\delta$, for some substitution $\delta$.

# Appendix B: Correctness of the Schemata for Semantics of MProlog in $KDI4_s$ and $KDI4$

In this appendix, we prove Theorem 6.1, which states that the given schemata for semantics of MProlog in $KDI4_s$ and $KDI4$ are correct. Let $L$ denote $KDI4_s$ or $KDI4$. Lemmas 4.4 and 4.10 – 4.13 and can be easily verified. The proofs of Theorem 4.3 and Lemma 4.7 are similar to the ones given for $BSMM$.

**Proof of Lemma 4.2** By the definition, $M$ is an $L$-model. Let $\{R'_i \mid 1 \leq i \leq m\}$ be the skeleton of $M$. We prove by induction on the length of $\alpha$ that for any $w \in W$, if $\alpha \in H(w)$, then $M, \sigma, w \vDash \alpha$. Let $w = \langle E_1 \rangle_{i_1} \ldots \langle E_k \rangle_{i_k}$ and $\triangle = w$ (be a modality).

The cases when $\alpha$ is a classical atom or $\alpha = \langle E \rangle_i \beta$ are trivial. Suppose that $\alpha = \Box_i \beta$. Let $u$ be a world such that $R_i(w,u)$ holds. By the inductive assumption, it suffices to show that $\beta \in H(u)$. Since $R_i(w,u)$ holds, there are two cases:

- Case $L = KDI4_s$, $w_0 = w$, $R'_{j_1}(w_0,w_1)$, $R'_{j_2}(w_1,w_2)$, $\ldots$, $R'_{j_h}(w_{h-1},w_h)$, $R'_j(w_h,u)$, $h \geq 0$ and $j \leq i$ : Since $\Box_i \beta \in H(w)$, by Lemma 4.1, there exists a $\Box$-lifting form $\triangle'$ of $\triangle$ such that $\triangle' \Box_i \beta \in Ext_L(I)$. By the definition of $Ext_L(I)$, it follows that $\triangle' \Box_{j_1} \ldots \Box_{j_h} \Box_j \beta \in Ext_L(I)$. Hence, by Lemma 4.1, $\beta \in H(u)$.
- Case $L = KDI4$, $w_0 = w$, $R'_{j_1}(w_0,w_1)$, $R'_{j_2}(w_1,w_2)$, $\ldots$, $R'_{j_h}(w_{h-1},w_h)$, $R'_j(w_h,u)$, $h \geq 0$ and $k \leq i$ for all $k \in \{j_1, \ldots, j_h, j\}$ : The assertion holds by the same argumentation as for the above case.

**Proof of Lemma 4.5** If $\alpha$ is of the form $\triangle E$ with $\triangle$ being a universal modality, then $\alpha \in Ext_L(I)$ (since $M \vDash \alpha$), and hence $\alpha \in Sat_L(I)$. Suppose that $\alpha = \triangle \Diamond_i E$, where $\triangle = \Box_{i_1} \ldots \Box_{i_k}$ with $k \geq 0$. Let $\langle W, \tau, R_1, \ldots, R_m, H \rangle$ be the standard $L$-model graph of $I$ and $\triangle' = w = \langle \top \rangle_{i_1} \ldots \langle \top \rangle_{i_k}$. Since $M \vDash \alpha$, we have $M, w \vDash \Diamond_i E$. There are two cases:

- Case $L = KDI4_s$, $E \in H(u)$, $u = w \langle F_1 \rangle_{j_1} \ldots \langle F_h \rangle_{j_h} \langle F \rangle_j$, $h \geq 0$, $j \leq i$ : By Lemma 4.1, some $\Box$-lifting form of $\triangle' \langle F_1 \rangle_{j_1} \ldots \langle F_h \rangle_{j_h} \langle F \rangle_j E$ belongs to $Ext_L(I)$. It follows that some $\Box$-lifting form of $\triangle \langle F_1 \rangle_{j_1} \ldots \langle F_h \rangle_{j_h} \langle F \rangle_j E$ belongs to $Sat_L(I)$. Hence $\triangle \Diamond_i E \in Sat_L(I)$.
- Case $L = KDI4$, $E \in H(u)$, $u = w \langle F_1 \rangle_{j_1} \ldots \langle F_h \rangle_{j_h} \langle F \rangle_j$, $h \geq 0$, $j_1 \leq i$, $\ldots$, $j_h \leq i$, $j \leq i$ : The assertion holds by a similar argumentation as for the above case.

**Proof of Lemma 4.8** Let $M$ be the standard $L$-model of $I$ and $\sigma$ the standard $\Diamond$-realization function on $M$. By the definition of $L$-instances of program clauses and the construction of $M$, it is sufficient to prove that for any ground $L$-instance $\triangle(A \leftarrow B_1, \ldots, B_n)$ of some program clause of $P$, for any $w \in W$ being an $L$-instance of $\triangle$, $M, w \vDash (A \leftarrow B_1, \ldots, B_n)$. Suppose that $M, w \vDash B_i$ for all $1 \leq i \leq n$. We show that $M, w \vDash A$.

Let $\triangle' = w$. We first show that for any ground simple atom $B$ of the form $E$, $\Box_i E$, or $\Diamond_i E$, if $M, w \vDash B$ then $\triangle' B$ is an $L$-instance of some atom from $Sat_L(I)$. If $B$ is of the form $E$ or $\Box_i E$, then by Lemma 4.1, some $\Box$-lifting form of $\triangle' B$ belongs to $Ext_L(I)$, and hence $\triangle' B$ is an $L$-instance of some atom from $Sat_L(I)$. Now consider the case when $B$ is of the form $\Diamond_i E$. There exists $u$ such that $R_i(w,u)$ and $M, u \vDash E$. There are two cases:

- Case $L = KDI4_s$, $u = w \langle F_1 \rangle_{j_1} \ldots \langle F_h \rangle_{j_h} \langle F \rangle_j$, $h \geq 0$, $j \leq i$ : By Lemma 4.1, some $\Box$-lifting form of $\triangle' \langle F_1 \rangle_{j_1} \ldots \langle F_h \rangle_{j_h} \langle F \rangle_j E$ belongs to $Ext_L(I)$. It follows that $\triangle' \langle F_1 \rangle_{j_1} \ldots \langle F_h \rangle_{j_h} \langle F \rangle_j E$ is an $L$-instance of some atom from $Sat_L(I)$, and hence so is $\triangle' \Diamond_i E$.
- Case $L = KDI4$, $u = w \langle F_1 \rangle_{j_1} \ldots \langle F_h \rangle_{j_h} \langle F \rangle_j$, $h \geq 0$, $j_1 \leq i$, $\ldots$, $j_h \leq i$, $j \leq i$ : The assertion holds by a similar argumentation as for the above case.

Since $M, w \vDash B_i$ for $1 \leq i \leq n$, it follows that $\triangle' B_i$ is an $L$-instance of some atom from $Sat_L(I)$. Consequently, $\triangle' A$ is an $L$-instance of some atom $\alpha$ from $T_{0L,P}(Sat_L(I))$. Since $T_{0L,P}(Sat_L(I)) = T_{L,P}(I) \subseteq I$, by Lemma 4.2, we have that $M, \sigma \vDash \alpha$, and hence $M, w \vDash A$.

## Appendix C: Correctness of the Schema for Semantics of $KDI45$-MProlog

In this appendix, we prove Theorem 6.3, which states that the schema for semantics of $KDI45$-MProlog given in Table 5 is correct. Let $L$ denote the logic $KDI45$. Lemmas 4.4 and 4.10 – 4.13 can be easily verified.

**Proof of Lemma 4.2** By the definition, $M$ is an $L$-model. Let $\{R_i \mid 1 \leq i \leq m'\}$ be the skeleton of $M$. We prove by induction on the length of $\alpha$ that for any $w \in W$, if $\alpha \in H(w)$, then $M, \sigma, w \vDash \alpha$. The cases when $\alpha$ is a classical atom or $\alpha = \langle E \rangle_i \beta$ are trivial. Suppose that $\alpha = \Box_i \beta$. Let $u$ be a world such that $R_i(w, u)$ holds. We show that $\beta \in H(u)$. Since $R_i(w, u)$ holds, there exist worlds $w_0, \ldots, w_k$ ($k \geq 0$) and $u_0, \ldots, u_h$ ($h \geq 1$) such that $w_0 = u_0$, $w_k = w$, $u_h = u$, $R'_{i_t}(w_{t-1}, w_t)$ for $1 \leq t \leq k$ with $i \geq i_1 > \ldots > i_k$, and $R'_{j_s}(u_{s-1}, u_s)$ for $1 \leq s \leq h$ with $i \geq j_1 > \ldots > j_h$. On the other side, because that all atoms of $H(w_0)$ are in the $L$-normal form and $\Box_i \beta \in H(w)$, we derive $k = 0$. Hence $\Box_i \beta \in H(u_0)$. By Lemma 4.1, there exists a $\Box$-lifting form $\triangle$ of $u_0$ such that $\triangle \Box_i \beta \in Ext_L(I)$. By the definition of $Ext_L$, it follows that $\triangle \Box_{j_1} \ldots \Box_{j_h} \beta \in Ext_L(I)$. Hence, by Lemma 4.1, $\beta \in H(u)$.

**Proof of Theorem 4.3** Let $M = \langle W, \tau, R_1, \ldots, R_m, H \rangle$ be the standard $L$-model graph of $I$, $\sigma$ the standard $\Diamond$-realization function and $\{R_i \mid 1 \leq i \leq m'\}$ the skeleton of the standard $L$-model of $I$. By Lemma 4.2, $M$ is an $L$-model of $I$. Let $M' = \langle W, \tau, R_1, \ldots, R_m, H' \rangle$ be the least extension of $M$ such that if $\Box_i \alpha \in H'(w)$, $R'_j(w, u)$ and $j \leq i$ then $\Box_i \alpha \in H'(u)$. Let $N = \langle D, W_2, \tau_2, S_1, \ldots, S_m, \pi \rangle$ be an arbitrary $L$-model of $I$ and $\sigma_2$ a $\Diamond$-realization function on $N$ such that $N, \sigma_2 \vDash I \cup \{\triangle \langle \top \rangle_i \mid \triangle$ is a universal modality$\}$.

Let $r \subseteq W \times W_2$ be the least relation such that, for all $w$, $w_2$, $u_2$, $E$, $i$:

- $r(\tau, \tau_2)$;
- if $r(w, w_2)$ and $R'_i(w, w\langle E \rangle_i)$ hold, and $\sigma_2(w_2, \langle E \rangle_i)$ is defined, then $r(w\langle E \rangle_i, \sigma_2(w_2, \langle E \rangle_i))$;
- if $r(w, w_2)$ and $S_i(w_2, u_2)$ hold and $w = \langle E_1 \rangle_{i_1} \ldots \langle E_k \rangle_{i_k}$ with $k \geq 0$, then $r(u, u_2)$ for $u = \langle E_1 \rangle_{i_1} \ldots \langle E_j \rangle_{i_j} \langle \top \rangle_i$ with $0 \leq j \leq k$ such that $i_j > i$ (if $j > 0$) and $i_{j+1} \leq i$ (if $j < k$).

The last condition in the above definition of $r$ implies that: if $r(w, w_2)$ and $S_i(w_2, u_2)$ hold then there exists $u$ such that $r(u, u_2)$ and $R_i(w, u)$.

We prove that $M' \leq_r N$. We first show that if $r(u, u_2)$ and $\alpha \in H'(u)$ then $N, \sigma_2, u_2 \vDash \alpha$. We prove this by induction on the step in the construction of $r$ at which $r(u, u_2)$ is created. Suppose that $r(u, u_2)$ holds and $\alpha \in H'(u)$. The case $u = \tau$ is trivial. There are two remaining cases:

- $u = w\langle E \rangle_i$, $u_2 = \sigma_2(w_2, \langle E \rangle_i)$, $r(w, w_2)$, and $R'_i(w, w\langle E \rangle_i)$; or
- $u = \langle E_1 \rangle_{i_1} \ldots \langle E_j \rangle_{i_j} \langle \top \rangle_i$, $w = \langle E_1 \rangle_{i_1} \ldots \langle E_k \rangle_{i_k}$, $0 \leq j \leq k$, $i_j > i$ (if $j > 0$), $i_{j+1} \leq i$ (if $j < k$), $r(w, w_2)$, and $S_i(w_2, u_2)$.

Consider the first case. Since $\alpha \in H'(u)$, there are three subcases: i) $\Box_i \alpha \in H'(w)$, ii) $\langle E \rangle_i \alpha \in H'(w)$, iii) $\alpha$ is of the form $\Box_j \beta$, $j \geq i$, and $\Box_j \beta \in H'(w)$. For the first two subcases, by the inductive assumption, either $N, \sigma_2, w_2 \vDash \Box_i \alpha$ or $N, \sigma_2, w_2 \vDash \langle E \rangle_i \alpha$, and hence $N, \sigma_2, \sigma_2(w_2, \langle E \rangle_i) \vDash \alpha$, which means that $N, \sigma_2, u_2 \vDash \alpha$. Consider the last subcase. By the inductive assumption, $N, \sigma_2, w_2 \vDash \Box_j \beta$, and hence $N, \sigma_2, w_2 \vDash \Box_i \Box_j \beta$ (by the axioms (4) and (I)). Therefore $N, \sigma_2, \sigma_2(w_2, \langle E \rangle_i) \vDash \Box_j \beta$, which means that $N, \sigma_2, u_2 \vDash \alpha$.

Consider the second case. Let $v$ be the world such that $u = v\langle \top \rangle_i$. Since $\alpha \in H'(u)$, either $\Box_i \alpha \in H'(v)$ or $\alpha$ is of the form $\Box_j \beta$, $j \geq i$, and $\Box_j \beta \in H'(v)$. Suppose that $\Box_i \alpha \in H'(v)$. It follows that $\Box_i \alpha \in H'(w)$. By the inductive assumption, $N, \sigma_2, w_2 \vDash \Box_i \alpha$, and hence $N, \sigma_2, u_2 \vDash \alpha$ since $S_i(w_2, u_2)$. Now suppose that $\alpha$ is of the form $\Box_j \beta$, $j \geq i$, and $\Box_j \beta \in H'(v)$. We have $\Box_j \beta \in H'(w)$. By the inductive assumption, $N, \sigma_2, w_2 \vDash \Box_j \beta$, and hence $N, \sigma_2, u_2 \vDash \Box_j \beta$ since $S_i(w_2, u_2)$ and $j \geq i$. This means that $N, \sigma_2, u_2 \vDash \alpha$.

We now show that if $r(w, w_2)$ and $R'_i(w, w\langle E \rangle_i)$ hold then $\sigma_2(w_2, \langle E \rangle_i)$ is defined. The case $E = \top$ is trivial. Suppose that $r(w, w_2)$ and $R'_i(w, w\langle E \rangle_i)$ hold, and $E \neq \top$. Thus, there exists

$\langle E\rangle_i\alpha \in H'(w)$ for some $\alpha$. By the first assertion, $N, \sigma_2, w_2 \vDash \langle E\rangle_i\alpha$. Hence $\sigma_2(w_2, \langle E\rangle_i)$ is defined. Therefore, the second condition in the above definition of $r$ can be simplified to "if $r(w, w_2)$ and $R'_i(w, w\langle E\rangle_i)$ hold then $r(w\langle E\rangle_i, \sigma_2(w_2, \langle E\rangle_i))$".

Next, we show that if $r(u, u_2)$ holds and $u = w\langle E\rangle_i$ then there exists $w_2$ such that $r(w, w_2)$ and $S_i(w_2, u_2)$. We prove this by induction on the construction of $r$. Suppose that $r(u, u_2)$ and $u = w\langle E\rangle_i$ hold. If $r(u, u_2)$ is created by the second condition of the definition of $r$, then the assertion clearly holds. Suppose that $r(u, u_2)$ is created by the third condition of the definition of $r$, i.e. $r(u, u_2)$ is created from $r(v, v_2)$ and the accompanying conditions that $u = w\langle\top\rangle_i$, $S_i(v_2, u_2)$, $v = w\langle E_1\rangle_{i_1}\ldots\langle E_k\rangle_{i_k}$, $k \geq 0$, $i \geq i_1 > i_2 > \ldots > i_k$. By the inductive assumption for $r(v, v_2)$, there exist $w_2$ such that $r(w, w_2)$ and a path from $w_2$ to $v_2$ consisting of edges connected via $S_{i_1}, \ldots, S_{i_k}$. Thus $S_i(w_2, u_2)$ holds.

To prove $M' \leq_r N$ it remains to show that if $r(w, w')$ and $R_i(w, u)$ hold, then there exists $u' \in W_2$ such that $r(u, u')$ and $S_i(w', u')$. Suppose that $r(w, w')$ and $R_i(w, u)$ hold. Since $R_i(w, u)$ holds, there exist worlds $w_0, \ldots, w_k$ ($k \geq 0$) and $u_0, \ldots, u_h$ ($h \geq 1$) such that $w_0 = u_0$, $w_k = w$, $u_h = u$, $R'_{i_t}(w_{t-1}, w_t)$ for $1 \leq t \leq k$ with $i \geq i_1 > \ldots > i_k$, and $R'_{j_s}(u_{s-1}, u_s)$ for $1 \leq s \leq h$ with $i \geq j_1 > \ldots > j_h$. Since $r(w, w')$ holds, by the assertion proved in the previous paragraph, there exists $w'_0$ such that $r(w_0, w'_0)$ and $S_i(w'_0, w')$. Hence, by the construction of $r$, there exists $u'$ such that $r(u, u')$ and $S_i(w'_0, u')$. Thus we have $r(u, u')$ and $S_i(w', u')$.

Therefore $M' \leq_r N$ and the standard $L$-model graph of $I$ is a least $L$-model of $I$.

**Proof of Lemma 4.5** If $\alpha$ is of the form $E$ or $\Box_i E$, then $\alpha \in Ext_L(I)$ (since $M \vDash \alpha$), and hence $\alpha \in Sat_L(I)$. Suppose that $\alpha = \Diamond_i E$. Let $\langle W, \tau, R_1, \ldots, R_m, H\rangle$ be the standard $L$-model graph of $I$. Since $M \vDash \alpha$, there exists a world $u = \langle E_1\rangle_{i_1}\ldots\langle E_k\rangle_{i_k}$ (with $k \geq 1$) of $M$ such that $i \geq i_1 > \ldots > i_k$ and $E \in H(u)$. By Lemma 4.1, some $\Box$-lifting form of $\langle E_1\rangle_{i_1}\ldots\langle E_k\rangle_{i_k}E$ belongs to $Ext_L(I)$, and hence also to $Sat_L(I)$. By the definition of $Sat_L$, it follows that there exists some $\nabla_j E \in Sat_L(I)$ with $j \leq i$. Hence $\alpha$ is an $L$-instance of some atom from $Sat_L(I)$.

**Proof of Lemma 4.7** Let $M = \langle D, W, \tau, R_1, \ldots, R_m, \pi\rangle$ be an arbitrary $L$-model of $P$ and $\sigma$ a maximal $\Diamond$-realization function on $M$. Note that if $M, \sigma \vDash \triangle\langle E\rangle_i E$ then $M, \sigma \vDash NF_L(\{\triangle\langle E\rangle_i E\})$. It is straightforward to prove by induction on $n$ that $M, \sigma \vDash T_{L,P}\uparrow n$. Therefore $M, \sigma \vDash I_{L,P}$ and $P \vDash_L I_{L,P}$.

**Proof of Lemma 4.8** Let $M$ be the standard $L$-model of $I$ and $\sigma$ the standard $\Diamond$-realization function on $M$. By the definition of $L$-instances of program clauses and the construction of $M$, it is sufficient to prove that for any ground $L$-instance $\triangle(A \leftarrow B_1, \ldots, B_n)$ of some program clause of $P$, for any $w \in W$ being an $L$-instance of $\triangle$, $M, w \vDash (A \leftarrow B_1, \ldots, B_n)$. Suppose that $M, w \vDash B_i$ for all $1 \leq i \leq n$. We show that $M, w \vDash A$.

Let $\triangle' = w$. We first show that for any ground simple atom $B$ of the form $E$, $\Box_i E$, or $\Diamond_i E$, if $M, w \vDash B$ then $\triangle' B$ is an $L$-instance of some atom from $Sat_L(I)$. Suppose that $M, w \vDash B$. If $B = E$, then by Lemma 4.1, some $\Box$-lifting form of $\triangle' B$ belongs to $Ext_L(I)$, and hence $\triangle' B$ is an $L$-instance of some atom from $Sat_L(I)$. Now suppose that $B = \Box_i E$. Let $w = \langle E_1\rangle_{i_1}\ldots\langle E_k\rangle_{i_k}$ and $0 \leq j \leq k$ be the index such that $i_j > i$ (if $j > 0$) and $i \geq i_{j+1}$ (if $j < k$). Let $u = \langle E_1\rangle_{i_1}\ldots\langle E_j\rangle_{i_j}\langle\top\rangle_i$. We have $R_i(w, u)$, and hence $M, u \vDash E$. By Lemma 4.1, it follows that some $\Box$-lifting form of $\langle E_1\rangle_{i_1}\ldots\langle E_j\rangle_{i_j}\Box_i E$ belongs to $Ext_L(I)$. Hence, by the definition of $Sat_L$, $\triangle' B$ is an $L$-instance of some atom from $Sat_L(I)$. Next, suppose that $B = \Diamond_i E$. Let $u$ be a world such that $R_i(w, u)$ and $M, u \vDash E$. Since $R_i(w, u)$ holds, there exists $\triangle''$, $\triangle^{(1)}$, $\triangle^{(2)}$ such that $w = \triangle''\triangle^{(1)}$, $u = \triangle''\triangle^{(2)}$, $\triangle''$ and $\triangle^{(1)}$ can be empty while $\triangle^{(2)}$ cannot, $\triangle^{(1)}$ and $\triangle^{(2)}$ contain only modal operators of kind (i.e. degree) less than or equal to $i$, while $\triangle''$ contains only modal operators of kind greater than $i$. Since $M, u \vDash E$, by Lemma 4.1, some $\Box$-lifting form of $\triangle''\triangle^{(2)}E$ belongs to $Ext_L(I)$. Hence, by the definition of $Sat_L$, there exists $\nabla_j$ with $j \leq i$ such that some $\Box$-lifting form of $\triangle''\nabla_j E$ belongs to $Sat_L(I)$. If $\triangle^{(1)}$ is not empty then, by the definition of $Sat_L$, it follows that some $\Box$-lifting form of $\triangle''\triangle^{(1)}\Diamond_i E$ belongs to $Sat_L(I)$. Hence $\triangle' B$ is an $L$-instance of some atom from $Sat_L(I)$.

Since $M, w \vDash B_i$ for $1 \leq i \leq n$, it follows that $\triangle' B_i$ is an $L$-instance of some atom from $Sat_L(I)$. Consequently, $\triangle' A$ is an $L$-instance of some atom $\alpha$ from $T_{0L,P}(Sat_L(I))$. Let $\alpha'$ be the $L$-normal form of $\alpha$, i.e. $NF_L(\{\alpha\}) = \{\alpha'\}$. We have $\alpha' \in T_{L,P}(I) \subseteq I$. By Lemma 4.2, $M, \sigma \vDash \alpha'$. From this we can conclude that $M, w \models A$.

# Appendix D: Correctness of the Schema for Semantics of $KD45_{(m)}$-MProlog

In this appendix, we prove Theorem 7.2, which states that the given schema for semantics of $KD45_{(m)}$-MProlog is correct. Let $L$ denote $KD45_{(m)}$. Lemmas 4.4 and 4.10 – 4.13 can be easily verified. The proof of Lemma 4.7 is the same as for $KDI45$.

**Proof of Lemma 4.2** By the definition, $M$ is an $L$-model. Let $\{R'_i \mid 1 \leq i \leq m\}$ be the skeleton of $M$. We prove by induction on the length of $\alpha$ that for any $w \in W$, if $\alpha \in H(w)$ then $M, \sigma, w \models \alpha$. The cases when $\alpha$ is a classical atom or $\alpha = \langle E \rangle_i \beta$ are trivial. Consider the remaining case when $\alpha = \Box_i \beta$. Let $u$ be a world such that $R_i(w,u)$ holds. Due to the $L$-normal form of modalities, $\Box_i \beta \in H(w)$ and $R_i(w,u)$ imply that $R'_i(w,u)$ holds, and hence $\beta \in H(u)$. By the inductive assumption, $M, \sigma, u \models \beta$. Hence $M, \sigma, w \models \alpha$.

**Proof of Theorem 4.3** Let $M = \langle W, \tau, R_1, \ldots, R_m, H \rangle$ be the standard $L$-model graph of $I$, $\sigma$ the standard $\Diamond$-realization function and $\{R'_i \mid 1 \leq i \leq m\}$ the skeleton of the standard $L$-model of $I$. By Lemma 4.2, $M$ is an $L$-model of $I$. Let $N = \langle D, W_2, \tau_2, S_1, \ldots, S_m, \pi \rangle$ be an arbitrary $L$-model of $I$ and $\sigma_2$ a $\Diamond$-realization function on $N$ such that $N, \sigma_2 \models I \cup \{\triangle \langle \top \rangle_i \mid \triangle$ is a universal modality$\}$.

Let $r \subseteq W \times W_2$ be the least relation such that, for all $w$, $w_2$, $u_2$, $E$, $i$:

- $r(\tau, \tau_2)$;
- if $r(w, w_2)$ and $R'_i(w, w\langle E \rangle_i)$ hold, and $\sigma_2(w_2, \langle E \rangle_i)$ is defined, then $r(w\langle E \rangle_i, \sigma_2(w_2, \langle E \rangle_i))$;
- if $r(w, w_2)$ and $S_i(w_2, u_2)$ hold then $r(u, u_2)$ holds for $u$ such that: if $w = \tau$ then $u = \langle \top \rangle_i$, if $w = v\langle E \rangle_j$ and $j \neq i$ then $u = w\langle \top \rangle_i$, and if $w = v\langle E \rangle_i$ then $u = v\langle \top \rangle_i$.

The last condition in the above definition of $r$ implies that: if $r(w, w_2)$ and $S_i(w_2, u_2)$ hold then there exists $u$ such that $r(u, u_2)$ and $R_i(w, u)$.

It is straightforward to prove by induction on the construction of $r$ that, if $R'_i(x, y)$ and $r(y, y_2)$ then there exists $x_2$ such that $r(x, x_2)$ and $S_i(x_2, y_2)$. Denote this assertion by (*).

We prove that $M \leq_r N$. We first show that if $r(u, u_2)$ and $\alpha \in H(u)$ then $N, \sigma_2, u_2 \models \alpha$. We prove this by induction on the length of $u$. The case $u = \tau$ is trivial. Suppose that $r(u, u_2)$, $u \neq \tau$, and $\alpha \in H(u)$. There are the following cases to consider:

- $u = w\langle E \rangle_i$, $u_2 = \sigma_2(w_2, \langle E \rangle_i)$, and $r(w, w_2)$ : Since $\alpha \in H(u)$, either $\Box_i \alpha \in H(w)$ or $\langle E \rangle_i \alpha \in H(w)$. Hence, by the inductive assumption, $N, \sigma_2, w_2 \models \Box_i \alpha$ or $N, \sigma_2, w_2 \models \langle E \rangle_i \alpha$. It follows that $N, \sigma_2, u_2 \models \alpha$.
- $u = \langle \top \rangle_i$ and $S_i(\tau_2, u_2)$ : Since $\alpha \in H(u)$, it follows that $\Box_i \alpha \in H(\tau)$. Hence $N, \sigma_2, \tau_2 \models \Box_i \alpha$ and $N, \sigma_2, u_2 \models \alpha$ (since $S_i(\tau_2, u_2)$).
- $u = w\langle \top \rangle_i$, $w = v\langle E \rangle_j$ with $j \neq i$, $r(w, w_2)$, and $S_i(w_2, u_2)$ : Since $\alpha \in H(u)$, we have $\Box_i \alpha \in H(w)$. Hence, by the inductive assumption, $N, \sigma_2, w_2 \models \Box_i \alpha$. It follows that $N, \sigma_2, u_2 \models \alpha$, since $S_i(w_2, u_2)$ holds.
- $u = v\langle \top \rangle_i$, $w = v\langle E \rangle_i$, $r(w, w_2)$, and $S_i(w_2, u_2)$ : By the assertion (*), there exists $v_2$ such that $r(v, v_2)$ and $S_i(v_2, w_2)$. Thus $S_i(v_2, u_2)$ holds. Since $\alpha \in H(u)$, we have $\Box_i \alpha \in H(v)$. By the inductive assumption, it follows that $N, \sigma_2, v_2 \models \Box_i \alpha$. Hence $N, \sigma_2, u_2 \models \alpha$.

Similarly as for the case of $BSMM$, it can be shown that if $r(w, w_2)$ and $R'_i(w, w\langle E \rangle_i)$ hold then $\sigma_2(w_2, \langle E \rangle_i)$ is defined. Therefore, the second condition in the above definition of $r$ can be simplified to "if $r(w, w_2)$ and $R'_i(w, w\langle E \rangle_i)$ hold then $r(w\langle E \rangle_i, \sigma_2(w_2, \langle E \rangle_i))$".

To prove $M \leq_r N$, it remains to show that if $r(w, w_2)$ and $R_i(w, u)$ then there exists $u_2 \in W_2$ such that $r(u, u_2)$ and $S_i(w_2, u_2)$. Suppose that $r(w, w_2)$ and $R_i(w, u)$ hold. We have that either

$u = w\langle E\rangle_i$ for some $E$ or there exists $v$ such that $w = v\langle E\rangle_i$ and $u = v\langle F\rangle_i$ for some $E$, $F$. If the first case occurs then choose $u_2 = \sigma_2(w, \langle E\rangle_i)$. Consider the second case. Since $w = v\langle E\rangle_i$ and $r(w, w_2)$, by the assertion (*), there exists $v_2$ such that $r(v, v_2)$ and $S_i(v_2, w_2)$. Since $r(v, v_2)$ and $u = v\langle F\rangle_i$, there exists $u_2$ such that $r(u, u_2)$ and $S_i(v_2, u_2)$. Thus, we have $r(u, u_2)$ and $S_i(w_2, u_2)$. This completes the proof that the standard $L$-model graph of $I$ is a least $L$-model of $I$.

**Proof of Lemma 4.5** Let $\langle W, \tau, R_1, \ldots, R_m, H\rangle$ be the standard $L$-model graph of $I$, $\triangle = \Box_{i_1} \ldots \Box_{i_k}$ be a modality, and $w = \langle\top\rangle_{i_1} \ldots \langle\top\rangle_{i_k}$. Suppose that $\alpha$ is of the form $\triangle E$. Since $M \vDash \alpha$, we have $M, w \vDash E$. Hence, by Lemma 4.1, $\triangle E \in Ext_L(I)$, and we also have $\triangle E \in Sat_L(I)$. Now suppose that $\alpha$ is of the form $\triangle\Diamond_i E$ with $i \neq i_k$. Since $M \vDash \alpha$, we have $M, w \vDash \Diamond_i E$ and there exists $\nabla_i$ such that $\nabla_i E \in H(w)$. Hence, by Lemma 4.1, $\triangle\nabla_i E \in Ext_L(I)$. Therefore $\triangle\Diamond_i E$ is an instance of some atom of $Sat_L(I)$.

**Proof of Lemma 4.8** Let $M$ be the standard $L$-model of $I$ and $\sigma$ the standard $\Diamond$-realization function on $M$. By the definition of $L$-instances of program clauses and the construction of $M$, it is sufficient to prove that for any ground $L$-instance $\triangle(A \leftarrow B_1, \ldots, B_n)$ of some program clause of $P$, for any $w \in W$ being an $L$-instance of $\triangle$, $M, w \vDash (A \leftarrow B_1, \ldots, B_n)$. Suppose that $M, w \vDash B_i$ for all $1 \leq i \leq n$. We show that $M, w \vDash A$.

Let $\triangle' = w = \langle E_1\rangle_{i_1} \ldots \langle E_k\rangle_{i_k}$. We first show that for any ground simple atom $B$ of the form $E$, $\Box_i E$, or $\Diamond_i E$, if $M, w \vDash B$ then $\triangle' B$ is an $L$-instance of some atom from $Sat_L(I)$. Suppose that $M, w \vDash B$. If $k = 0$ or $i_k \neq i$, then let $v = w$, else let $v = \langle E_1\rangle_{i_1} \ldots \langle E_{k-1}\rangle_{i_{k-1}}$. If $B = E$, then by Lemma 4.1, some $\Box$-lifting form of $\triangle' B$ belongs to $Ext_L(I)$, and hence $\triangle' B$ is an $L$-instance of some atom from $Sat_L(I)$. Now suppose that $B = \Box_i E$. Let $u = v\langle\top\rangle_i$ and $\triangle'' = v\Box_i$. We have $R_i(w, u)$, and hence $M, u \vDash E$. By Lemma 4.1, it follows that some $\Box$-lifting form of $\triangle'' E$ belongs to $Ext_L(I)$. Hence, by the definition of $Sat_L$, $\triangle' B$ is an $L$-instance of some atom from $Sat_L(I)$. Next, suppose that $B = \Diamond_i E$. Since $M, w \vDash B$, there exists $F$ such that $v\langle F\rangle_i$ is a world of $M$ and $M, v\langle F\rangle_i \vDash E$. Let $\triangle'' = v\langle F\rangle_i$. By Lemma 4.1, some $\Box$-lifting form of $\triangle'' E$ belongs to $Ext_L(I)$. Hence, by the definition of $Sat_L$, $\triangle' B$ is an $L$-instance of some atom from $Sat_L(I)$.

Since $M, w \vDash B_i$ for $1 \leq i \leq n$, it follows that $\triangle' B_i$ is an $L$-instance of some atom from $Sat_L(I)$. Consequently, $\triangle' A$ is an $L$-instance of some atom $\alpha$ from $T_{0L,P}(Sat_L(I))$. Let $\alpha'$ be the $L$-normal form of $\alpha$, i.e. $NF_L(\{\alpha\}) = \{\alpha'\}$. We have $\alpha' \in T_{L,P}(I) \subseteq I$. By Lemma 4.2, $M, \sigma \vDash \alpha'$. From this we can conclude that $M, w \vDash A$.

# Contents