On Modal Deductive Databases

Linh Anh Nguyen

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

ADBIS'2005, 12 September

Motivation

Modal and temporal logics are useful and it is desirable to study extensions of deductive databases using such logics.

Temporal deductive databases have received a lot of attentions from researchers. On the other hand, the term "modal deductive databases" is hard to find in the literature of computer science.

The aim of this work is to study theoretical aspects of modal deductive databases such as query languages, data complexity, and computational methods.

Related Work

- *Constructing the Least Models for Positive Modal Logic Programs*. Fundamenta Informaticae, 42(1) 2000, 29-60.

- *The Modal Query Language MDatalog*. Fundamenta Informaticae, 46(4), 2001, 315-342.

- *A Fixpoint Semantics and an SLD-Resolution Calculus for Modal Logic Programs*. Fundamenta Informaticae, 55(1), 2003, 63-100.

- *Multimodal Logic Programming and Its Applications to Modal Deductive Databases*. Manuscript, available at `http://www.mimuw.edu.pl/~nguyen/papers.html`. Published partially in the proceedings of ICLP'2004, JELIA'2004, CLIMA-VI, ICTAC'2005, and ADBIS'2005.

Overview

- Multimodal Logics of Belief
- Modal Logic Programs
- Fixpoint Semantics of Modal Logic Programs
- MDatalog and Modal Deductive Databases
- Modal Relational Algebras
- Bottom-up Computational Methods
- The Data Complexity of MDatalog

Multimodal Logics of Belief

Multimodal logics extend classical logic with modal operators $\Box_i$ and $\Diamond_i$, where $1 \leq i \leq m$ for some constant m.

For reasoning about epistemic states of agents:

$\Box_i\varphi$ stands for "agent i knows/believes that φ is true",

$\Diamond_i\varphi$ stands for "agent i considers that φ is possible".

For reasoning about multi-degree belief:

$\Box_i\varphi$ stands for "φ is believed up to degree i",

$\Diamond_i\varphi$ stands for "φ is possible weakly at degree i".

We use Kripke semantics with constant domain and rigid terms.

The satisfaction relation is defined w.r.t. the actual world of the model, i.e. $M \models \varphi$ if $M, \tau \models \varphi$, where τ is the actual world of M.

Multimodal Logics of Belief (2)

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

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

For reasoning about multi-degree belief we can use

$$KDI4_s5 = K_{(m)} + (D) + (I) + (4_s) + (5),$$
$$KDI45 = K_{(m)} + (D) + (I) + (4) + (5).$$

For distributed systems of belief: $KD4_s5_s = K_{(m)} + (D) + (4_s) + (5_s)$.

For reasoning about agents: $KD45_{(m)} = K_{(m)} + (D) + (4) + (5)$.

A Restriction for This Talk

In the paper, L can be any one of the four mentioned modal logics.

For simplicity, we will give details only for $KDI4_s5$.

Modal Logic Programs

A program clause is a formula of the form $\boxdot(A \leftarrow B_1, \ldots, B_n)$, where $\boxdot$ is a sequence of universal modal operators (called the *modal context*), $A, B_1, \ldots, B_n$ are of the form E, $\Box_i E$, or $\Diamond_i E$ with E being a classical atom.

An MProlog program is a finite set of program clauses.

In $L = KDI4_s5$ we have $\nabla\nabla'\varphi \equiv \nabla'\varphi$, where ∇ and ∇' are modal operators. Hence define that an MProlog program is an $KDI4_s5$-MProlog program if its program clauses have modal context with length bounded by 1.

Labeled Modal Operators

We use classical atoms to label existential modal operators $\Diamond_i$.

The forward labeled form of $\triangle\Diamond_i E$ is $\triangle\langle E\rangle_i E$,
where $\triangle$ is a sequence of modal operators and E is a classical atom.

The intuition is that:

$\Diamond_i$ says that there exists *some* possible world . . .
So, we can fix this world by labeling it by something.

Furthermore, we don't have $\Diamond_i A \wedge \Diamond_i B \rightarrow \Diamond_i(A \wedge B)$, but we have

$$\langle S\rangle_i A \wedge \langle S\rangle_i B \rightarrow \langle S\rangle_i(A \wedge B)$$

Normal Form of Modalities

A modality, denoted by $\triangle$, is a sequence of modal operators.

$\triangle$ is in labeled form if it does not contain unlabeled $\Diamond_i$.

$\triangle$ is in $KDI4_s5$-normal form if its length is 0 or 1.

$\triangle$ is in L-normal labeled form if it is in L-normal form and labeled form.

Model Generators

An atom is a formula of the form $\triangle E$,
where $\triangle$ is a modality and E is a classical atom.
It is in *L*-normal labeled form if $\triangle$ is in *L*-normal labeled form.

An atom is in almost *L*-normal labeled form if it is of the form $\triangle A$ with $\triangle$ in *L*-normal labeled form and A being a simple atom of the form E, $\Box_i E$, $\Diamond_i E$, or $\langle F \rangle_i E$ with E and F being classical atoms.

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

Ordering Modal Operators

Define $\preceq_L$ to be a pre-order between modal operators such that

$$\Diamond_i \preceq_L \langle S \rangle_i \preceq_L \Box_i$$

and if $L \in \{KDI4_s5, KDI45\}$ and $i < j$ then $\Box_i \preceq_L \Box_j$ and $\Diamond_j \preceq_L \Diamond_i$.

The above order can be extended for comparing modalities and we can specify when an atom is an L-instance of another.

For example, $\Box_1 \Diamond_2 E$ is a $KDI4_s5$-instance of $\Box_2 \langle F \rangle_1 E$.

Instances of Program Clauses

Let $\boxdot$ be a universal modality in L-normal form,
$\boxdot'$ a modal context of an L-MProlog program clause,
φ and φ' be program clauses with an empty modal context.

Define that:

$\boxdot$ is an *L-context instance* of $\boxdot'$ if $\boxdot'\psi \rightarrow \boxdot\psi$ is L-valid (for every ψ),

$\boxdot\varphi$ is an *L-instance of (a program clause)* $\boxdot'\varphi'$ if $\boxdot$ is an L-context instance of $\boxdot'$ and there exists a substitution θ such that $\varphi = \varphi'\theta$.

For example, $\Box_1(p(a) \leftarrow \Diamond_2 q(a))$ is an $KDI4_s5$-instance of the clause $\Box_2(p(x) \leftarrow \Diamond_2 q(x))$.

Fixpoint Semantics of MProlog Programs (1)

Fixpoint semantics of *L*-MProlog programs are defined using a direct consequence operator denoted by $T_{L,P}$. 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 $T_{0L,P}$.

- 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 $NF_L(T_{0L,P}(Sat_L(I)))$.

Fixpoint Semantics of MProlog Programs (2)

The saturation operator Sat_L for $L = KDI4_s5$ is specified by three rules: a) $\Box_i E \rightarrow \Box_j E$ if $i > j$, b) $\Box_i E \rightarrow \Box_m \Box_i E$, c) $\langle F \rangle_i E \rightarrow \Box_m \Diamond_i E$, where E and F are classical atoms.

Given an L-normal model generator I, $Sat_L(I)$ is the least extension of I that contains all ground atoms in almost L-normal labeled form that are derivable from some atom in I using the rules specifying Sat_L.

As an example, for $L = KDI4_s5$, we have
$Sat_L(\{\Box_2 p(a)\}) = \{\Box_2 p(a), \Box_1 p(a), \Box_m \Box_2 p(a), \Box_m \Box_1 p(a)\}$.

The normalization operator NF_L for $L = KDI4_s5$ is specified by the only rule $\nabla' \nabla E \rightarrow \nabla E$, where ∇ and ∇' are modal operators and E is a classical atom.

Given a set of atoms I, $NF_L(I)$ is the set of all ground atoms in L-normal labeled form that are derivable from some atom of I using the rules specifying NF_L.

Fixpoint Semantics of MProlog Programs (3)

For a set I of ground atoms in almost L-normal labeled form, $T_{0L,P}(I)$ is the least set of atoms such that if

- $\boxdot(A \leftarrow B_1, \ldots, B_n)$ is a ground L-instance of some program clause of P
- and $\triangle$ is a maximally general ground modality in L-normal labeled form such that $\triangle$ is an L-instance of $\boxdot$ 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 example, if $L = KDI4_s5$ and P consists of the only clause

$$\Box_2(\Diamond_1 p(x) \leftarrow q(x), r(x), \Box_1 s(x), \Diamond_2 t(x))$$

and $I = \{\langle q(a)\rangle_1 q(a),\ \langle q(a)\rangle_1 r(a),\ \Box_2\Box_2 s(a),\ \Box_2\langle t(a)\rangle_1 t(a)\}$,
then $T_{0L,P}(I) = \{\langle q(a)\rangle_1 \langle p(a)\rangle_1 p(a)\}$. Here, $\triangle = \langle q(a)\rangle_1$.

Fixpoint Semantics of MProlog Programs (4)

Recall that $T_{L,P}(I)$ is defined as $NF_L(T_{0L,P}(Sat_L(I)))$.

This operator $T_{L,P}$ is monotonic and continuous and has the least fixpoint

$$T_{L,P}\uparrow\omega = \bigcup_{n=0}^{\omega} T_{L,P}\uparrow n$$

where $T_{L,P}\uparrow 0 = \emptyset$ and $T_{L,P}\uparrow n = T_{L,P}(T_{L,P}\uparrow(n-1))$ for $n > 0$.

It is proved in another paper that the "standard *L*-model" of the least fixpoint $T_{L,P}\uparrow\omega$ is a "least *L*-model of P".

MDatalog

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 program free from function symbols and containing only allowed clauses.

Modal Deductive Databases

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 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.

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

A modal deductive database in *L* consists of an instance I of extentional *L*-relations (*edb*) and an *L*-MDatalog program P for defining intentional relations (*idb*).

An *L*-MDatalog program P can be treated as the function 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 $T_{L,P}(I \cup J) = J$.

Queries

An *L*-MDatalog query is a pair (P, φ), where P is an *L*-MDatalog program and $\varphi = \boxdot(query(x_1, \ldots, x_k) \leftarrow B_1, \ldots, B_h)$ is an *L*-MDatalog clause s.t.: $query$ is a special predicate symbol not occurring in P and the body of φ, the variables $x_1, \ldots, x_k$ are different, and $k \geq 1$.

An *L*-MDatalog query (P, φ) takes as input an instance I of *edb* *L*-relations and returns as output the *L*-relation $P'_L(I)(query)$, where $P' = P \cup \{\varphi\}$.

Example

Let us consider the situation when a company has some branches and a central database.

Each of the branches can access and update the database, and suppose that the company wants to distinguish data and knowledge coming from different branches.

Also assume that data coming from branches can contain noises and statements expressed by a branch may not be highly recognized by other branches. This means that data and statements expressed by branches are treated as "belief" rather than "knowledge".

In this case, we can use the multimodal logic $KD4_s5_s$, where each modal index represent a branch of the company, also called an *agent*. Recall that in this logic each agent has full access to the belief bases of the other agents.

Example (continued)

Data put by agent i are of the form $\Box_i E$ (agent i believes in E) or $\Diamond_i E$ (agent i considers that E is possible).

A statement expressed by agent i is a clause of the form $\Box_i(A \leftarrow B_1, \ldots, B_n)$, where A is an atom of the form E, $\Box_i E$, or $\Diamond_i E$, and $B_1, \ldots, B_n$ are simple modal atoms that may contain modal operators of the other agents.

For communicating with normal users, the central database may contain rules with the empty modal context, i.e. in the form $E \leftarrow B_1, \ldots, B_n$, which hide sources of information.

Example (continued)

As a concrete example, consider the following program/database in $KD4_s5_s$:

agent 1:

$\Box_1(\Diamond_1 likes(x, Coca) \leftarrow likes(x, Pepsi))$ (1)

$\Box_1(\Diamond_1 likes(x, Pepsi) \leftarrow likes(x, Coca))$ (2)

$\Box_1 likes(Tom, Coca) \leftarrow$ (3)

$\Box_1 likes(Peter, Pepsi) \leftarrow$ (4)

agent 2:

$\Box_2(likes(x, Coca) \leftarrow likes(x, Pepsi))$ (5)

$\Box_2(likes(x, Pepsi) \leftarrow likes(x, Coca))$ (6)

$\Box_2 likes(Tom, Pepsi) \leftarrow$ (7)

$\Box_2 likes(Peter, Coca) \leftarrow$ (8)

$\Box_2 likes(Peter, beer) \leftarrow$ (9)

agent 3:

$\Box_3(very_much_likes(x, y) \leftarrow likes(x, y), \Box_1 likes(x, y), \Box_2 likes(x, y))$ (10)

$\Box_3 likes(Tom, Coca) \leftarrow$ (11)

$\Diamond_3 likes(Peter, Pepsi) \leftarrow$ (12)

$\Diamond_3 likes(Peter, beer) \leftarrow$ (13)

for communicating with users:

$very_much_likes(x, y) \leftarrow \Box_3 very_much_likes(x, y)$ (14)

$likes(x, y) \leftarrow \Diamond_3 very_much_likes(x, y)$ (15)

$possibly_likes(x, y) \leftarrow \Diamond_i likes(x, y)$ (for $i \in \{1, 2, 3\}$) (16)

Modal Relational Algebras

We first define a modal relational algebra in L, called the L-SPCU algebra. These algebras extend the classical SPCU algebra with some operators involving with modalities.

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

- selection, projection, cross-product, union,
- context-shrink ($\Box_i$ and $\Diamond_i$), context-stretch ($\Box^{\leftarrow}$ and $\Diamond^{\leftarrow}$), context-selection ($\sigma_{\boxdot}$),
- saturation (Sat_L), labeling ($Label_p$), normalization (NF_L).

Modal Relational Algebras (2)

For example:

$I \times J = \{(\triangle, \langle \mathfrak{t}(1), \ldots, \mathfrak{t}(k), \mathfrak{s}(1), \ldots, \mathfrak{s}(h)\rangle) \mid$ there exist $\triangle'$ and $\triangle''$ such that $(\triangle', \mathfrak{t}) \in I$, $(\triangle'', \mathfrak{s}) \in J$, and $\triangle$ is a maximal *L*-instance in *L*-normal labeled form of $\triangle'$ and $\triangle''\}$.

$\Box_i(I) = \{(\triangle, \mathfrak{t}) \mid$ there exists $(\triangle\nabla, \mathfrak{t}) \in I$ such that $\Box_i \preceq_L \nabla\}$.

$\Box_i^{\leftarrow}(I) = \{(\triangle\Box_i, \mathfrak{t}) \mid (\triangle, \mathfrak{t}) \in I\}$

$\sigma_{\boxdot}(I) = \{(\triangle, \mathfrak{t}) \mid$ there exist $(\triangle', \mathfrak{t}) \in I$ and a universal modality $\boxdot'$ being an *L*-context instance of $\boxdot$ such that $\triangle$ is a maximal *L*-instance in *L*-normal labeled form of $\triangle'$ and $\boxdot'\}$.

$Label_p(I) = \{(\triangle, \mathfrak{t}) \mid (\triangle, \mathfrak{t}) \in I$ and $\triangle$ is not of the form $\triangle'\Diamond_i\} \cup$
$\{(\triangle\langle p(c_1, \ldots, c_n)\rangle_i, \langle c_1, \ldots, c_n\rangle) \mid (\triangle\Diamond_i, \langle c_1, \ldots, c_n\rangle) \in I\}$.

L-SPCU Queries

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

Theorem: Every *L*-MDatalog query (P, φ), where $L \in \{KDI4_s5,\ KDI45,\ KD4_s5_s,\ KD45_{(m)}\}$ and P is a nonrecursive *L*-MDatalog program, is equivalent to an *L*-SPCU query.

Example

Consider the following MDatalog program clause

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

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

$$NF_L(Label_{ans}(\Diamond_i^{\leftarrow}(\sigma_{\boxdot}(\ \sigma_{1=2}(Q \times Q) \times Sat_L(T) \times I_L^a\ ))))$$

Bottom-up Computational Methods

We have extended the seminaive evaluation and the magic-set transformation of Datalog for MDatalog. See the paper for details. One of the main results is that the magic-set transformation for MDatalog is correct.

Here are some specific points of the adaptation:

- Modal contexts of program clauses remain through transformations.

- Because $\boxdot(\Box_i E \rightarrow \Diamond_i E)$ holds, we implicitly adopt $\boxdot(\Diamond_i input_E \rightarrow \Box_i input_E)$.

- Because rules specifying operators Sat_L (saturation) and NF_L (normalization) are meta-clauses, we adopt appropriate clauses for defining "input" relations, similarly as in the above item.

The Data Complexity of MDatalog

Theorem: For $L \in \{KDI4_s5,\ KDI45,\ KD4_s5_s,\ KD45_{(m)}\}$, the data complexity of L-MDatalog is in PTIME.

Unpublished Results: The data complexity of L-MDatalog for

- $L \in \{KD, T, KB, KDB, B\}$ is complete in PTIME,
- $L \in \{K4, KD4, S4\}$ is complete in PSPAPE,
- $L = K$ is complete in coNP,
- $L \in \{K5, KD5, K45, KD45, KB5, S5\}$ is complete in PTIME (this case immediately follows from the existing results).

Conclusions and Future Work

We have established a fundamental basis for the subject of modal deductive databases.

Future work:

- Try to find practical applications of modal deductive databases.

- Instead of modal logics we will look at description logics.