PUBLICATIONS DE L'INSTITUT MATHÉMATIQUE
Nouvelle série, tome 82(96) (2007), 25–35 DOI 102298/PIM0796025M

DISJUNCTION IN MODAL DESCRIPTION LOGICS

Milenko Mosurović

Abstract. We investigate the complexity of satisfaction problems in modal description logics without disjunction between formulae. It is shown that simulation of disjunction in the class of all models of these logics is possible, so that the complexity remains same no matter the logics is with or without disjunction of formulae. However, the omission of disjunction, in the class of the models based on the universal relation, "turns down" the complexity of satisfaction problem i.e., if P $\neq$ NP, it is not possible to simulate disjunction.

1. Introduction

Description logics are invented for knowledge representation and reasoning in systems of artificial intelligence (see e.g. [**6, 5, 1**] and [**8**] for more references). An apparent general requirement to such logics is "to be sufficiently expressive and effective." However, the concrete balance between their expressive power and complexity depends on the application domain the logic is designed for. There is a wide spectrum of description logics, from relatively weak ones, like $ALER$, the (un)satisfaction problem for concepts in which is NP-complete, more complex ALC which is PSPACE-complete (see [**7**]), to very expressive ones, like CIQ of De Giacomo and Lenzerini [**10**] and De Giacomo [**9**].

The conventional description logics were designed to represent knowledge about static application domains only. To capture various dynamic features, for instance, intensional knowledge (in multi-agent systems), dependence on time or actions (in distributed systems), description logics are combined with suitable "modal" (propositional) logics, say epistemic, temporal, or dynamic. Again, there is a variety of possible combinations (see e.g [**15, 12, 2, 3**]). Some of them are rather simple and do not increase substantially the complexity of the combined logic (for example, the temporal description logic of Schild [**15**] is EXPTIME-complete); others are too expressive and undecidable (e.g. the multi-dimensional description logic of Baader and Ohlbach [**3**]).

2000 *Mathematics Subject Classification*: Primary 03B70; Secondary 68T27, 68Q17.

Various kinds of balances between the expressive power and decidability have been found in the series of papers [**16, 17, 18, 19**], where expressive and yet decidable description logics with epistemic, temporal, and dynamic operators were constructed. However, the complexity of the satisfaction problem in almost all of these logics is NEXPTIME-hard [**14**](some of these logics are NEXPTIME-complete e.g. [**17**] and some EXPSPACE-complete e.g. [**18**]).

The disjunction is usually source of non-determinism, and we recall that in modal description logics it can arise between concepts and between formulae. It is possible that the presence of disjunction of formulae cause such levels of complexity. The syntax of logics ALC_M of Baader and Laux does not have disjunction of formulae, so that we can expect the lower complexity for them.

In this paper we show that disjunction of ALC_M-formulae of Wolter and Zakharyaschev can be simulated in ALC_M-formulae of Baader and Laux in the class K and the satisfiability problem for ALC_M-formulae of Baader and Laux in the class K is NEXPTIME-hard. On the other hand, ALC_M-formulae of Baader and Laux in the class $S5$ is EXPTIME-complete (i.e. assuming P $\neq$ NP, the disjunction of ALC_M-formulae of Wolter and Zakharyaschev can not be simulated in ALC_M-formulae of Baader and Laux in the class $S5$).

2. Syntax and Semantics

We begin by defining the modal concept description language ALC_M and its semantics. The primitive symbols of ALC_M are:

concept names $C_0, C_1, \dots,$

role names $R_0, R_1, \dots,$ and

object names $a_0, a_1, \dots.$

Starting from primitive symbols, we can form compound concepts and formulae using the following constructs. Suppose R is a role name and C, D are concepts (for the basis of our inductive definition we assume concept names to be atomic concepts). Then $\top$, $C \wedge D$, $\neg C$, $\exists R.C$, and $\Diamond C$ are *concepts*.

Atomic formulae are expressions of the form $\top$, $C = D$, $a : C$, and aRb, where a, b are object names. If φ and ψ are formulae, then so are $\varphi \wedge \psi$, $\neg\varphi$, and $\Diamond\varphi$.

The corresponding modal description language is denoted by ALC_M.

Other standard logical connectives are defined in the usual way. For instance, $C \vee D$ is an abbreviation for $\neg(\neg C \wedge \neg D)$, $\bot$ for $\neg\top$,$C \to D$ for $\neg(C \wedge \neg D)$, $C \subseteq D$ for $C \wedge D = C$, and $\Box$ for $\neg\Diamond\neg$.

Note that, in the definition above, we did not impose any restriction on the form of conceptual assertional axioms. Baader and Laux [**2**] consider only atomic formulae prefixed by sequences of modal operators.

We recall to the syntax of ALC_{M^B} introduced by Baader and Laux [**2**].

Terminological axioms of ALC_{M^B} are of the form $m(C = D)$ where C and D are concepts of ALC_M and m is a (possibly empty) sequence of modal operators. Assertional axioms of ALC_{M^B} are of the form $m(aRb)$ or $m(a : C)$ where a and b are object names, R is a role name, C is a concept, and m is a (possibly empty) sequence of modal operators ($\Diamond$ and $\Box$). An ALC_{M^B}-formula is either a terminological or an assertional axiom.

A *model* of ALC_M based on a frame $\mathfrak{F} = \langle W, \lhd \rangle$ is a pair $\mathfrak{M} = \langle \mathfrak{F}, I \rangle$ in which I is a function associating with each $w \in W$ a structure

$$I(w) = \left\langle \Delta^{I,w}, R_0^{I,w}, \dots, C_0^{I,w}, \dots, a_0^{I,w}, \dots \right\rangle,$$

where $\Delta^{I,w}$ is a nonempty set of objects, the *domain* of w, $R_i^{I,w}$ are binary relations on $\Delta^{I,w}$, $C_i^{I,w}$ are subsets of $\Delta^{I,w}$, and $a_i^{I,w}$ are objects in $\Delta^{I,w}$ such that $a_i^{I,w} = a_i^{I,v}$, for any $v, w \in W$.

One can distinguish between three types of models: those with *constant*, *expanding*, and *varying domains*. In models with constant domains $\Delta^{I,v} = \Delta^{I,w}$, for all $v, w \in W$. In models with expanding domains $\Delta^{I,v} \subseteq \Delta^{I,w}$ whenever $v \lhd w$. And models with varying domains are just arbitrary models.

Given a model $\mathfrak{M}$ and a world w in it, we define the *value* $C^{I,w}$ of a concept C in w and the *satisfaction relation* $(\mathfrak{M}, w) \models \varphi$ (or simply $w \models \varphi$, if $\mathfrak{M}$ is understood) by taking:

$$\begin{aligned}
\top^{I,w} &= \Delta^{I,w} \ \text{ and} \\
C^{I,w} &= C_i^{I,w}, \ \text{ for } \ C = C_i; \\
(C \wedge D)^{I,w} &= C^{I,w} \cap D^{I,w}; \\
(\neg C)^{I,w} &= \Delta^{I,w} - C^{I,w}; \\
x \in (\Diamond C)^{I,w} & \ \text{ iff } \ \exists v \rhd w \ x \in C^{I,v}; \\
x \in (\exists R.C)^{I,w} & \ \text{ iff } \ \exists y \in C^{I,w} \ x R^{I,w} y; \\
w \models C = D & \ \text{ iff } \ C^{I,w} = D^{I,w}; \\
w \models a : C & \ \text{ iff } \ a^{I,w} \in C^{I,w}; \\
w \models aRb & \ \text{ iff } \ a^{I,w} R^{I,w} b^{I,w}; \\
w \models \Diamond \varphi & \ \text{ iff } \ \exists v \rhd w \ v \models \varphi; \\
w \models \varphi \wedge \psi & \ \text{ iff } \ w \models \varphi \ \text{ and } \ w \models \psi; \\
w \models \neg \varphi & \ \text{ iff } \ w \not\models \varphi.
\end{aligned}$$

A formula φ is *satisfiable* in a class of models M if there is a model $\mathfrak{M} \in M$ and a world w in $\mathfrak{M}$ such that $w \models \varphi$. Usually, we consider following classes of models:

- K the class of all models;
- $S5$ the class of models based on frames with the universal relations, i.e., $u \lhd v$ for all u and v;
- $KD45$ the class of transitive, serial ($\forall u \exists v \ u \lhd v$) and Euclidean ($u \lhd v \wedge u \lhd w \rightarrow v \lhd w$) models;
- $S4$ the class of all quasi-ordered models;
- $K4$ the class of transitive models;
- GL the class of transitive Noetherian models (i.e., containing no infinite ascending chains);
- N the class of models based on $\langle \mathbb{N}, < \rangle$.

It is obvious that finite conjunction of ALC_{M^B}-formulae is a ALC_M-formula.

A set $\{F_1, \dots, F_n\}$ of ALC_{M^B}-formulae is *satisfiable* in a class of models M if there is a model $\mathfrak{M} \in M$ and a world w in $\mathfrak{M}$ such that $w \models F_i$, for $i = 1, \dots, n$.

3. Complexity in the class K

Lower bounds of the satisfaction problem for some modal description logics with constant domain assumption follows from [**14**] and [**13**]. For instance, *the satisfaction problem for ALC_M-formulae free from role names in each of the classes K, $S4$, and $K4$ is* NEXPTIME*-hard.*

Now we will establish the lower bound for the satisfaction problem in the class K with expanding domain assumption.

THEOREM 3.1. *Testing satisfiability of a finite set $\{F_1, \dots, F_n\}$ of ALC_{M^B}-formulas in a class K with expanding domains is* NEXPTIME*-hard.*

The key step in the proof of Theorem 3.1 lies in showing that disjunction of ALC_M-formulae can be simulated in ALC_{M^B} in class K.

PROPOSITION 3.1. *Let $\mathfrak{M} = (W, <, I)$ be a model of ALC_M, let w be a world in W, and let E, F are concept names. If $\mathfrak{M}, w \models (F = \Diamond E) \wedge \Box(E = \top)$, then $\mathfrak{M}, w \models (F = \top) \vee (F = \bot)$.*

PROOF. Let $\mathfrak{M}, w \models (F = \Diamond E) \wedge \Box(E = \top)$. If $\exists v \in W w < v$, then $F^{I,w} = (\Diamond E)^{I,w} = \triangle^{I,w}$ (i.e., $\mathfrak{M}, w \models F = \top$), else $F^{I,w} = (\Diamond E)^{I,w} = \emptyset$ (i.e., $\mathfrak{M}, w \models F = \bot$). $\Box$

PROOF. (Theorem 3.1) We will show here the lower bound for the testing satisfiability of a finite set of ALC_{M^B}-formulae in a class K with expanding domains by reducing to it the $n \times n$ tiling problem, n given in binary, which is known to be NEXPTIME-complete [**4**]. Namely, for a set $T = \{t_1, \dots, t_s\}$ of tiles and $n < \omega$, we construct a finite set of ALC_{M^B}-formulae $F = \{\varphi_1, \dots, \varphi_m\}$ such that F is satisfied in an ALC_M-model from K iff T tiles $2^n \times 2^n$.

To encode the $2^n \times 2^n$ grid, we define 2^{2n} concepts B_{ij}, $0 \leqslant i, j < 2^n$, using $2n$ concept names $C_0, \dots, C_{2n-1}$, a role name R, and an object name a.

Let F_1 be the set of the following formulae (that are similar to those of [**11**, p. 371] and [**14**]):

$$\exists R.\top = \top, \quad a : \neg C_0 \wedge \dots \wedge \neg C_{2n-1},$$

$$\bigwedge_{j=0}^{i-1} C_j \to (C_i \to \forall R.\neg C_i) \wedge (\neg C_i \to \forall R.C_i) = \top, \text{ for } i = 0, \dots, 2n-1,$$

$$\bigvee_{j=0}^{i-1} \neg C_j \to (C_i \to \forall R.C_i) \wedge (\neg C_i \to \forall R.\neg C_i) = \top, \text{ for } i = 0, \dots, 2n-1.$$

For any $i, j \in \{0, \dots, 2^n - 1\}$ written in binary as $(d_{2n-1}, \dots, d_n)$ and $(d_{n-1}, \dots, d_0)$, respectively, we put $B_{ij} = C_0^{d_0} \wedge \dots \wedge C_{2n-1}^{d_{2n-1}}$, where C^d is C if $d = 1$ and $\neg C$ otherwise. If F_1 is satisfied in a world w in an ALC-model, then the sets B_{ij} in this model are nonempty, pairwise disjoint and cover the domain $\triangle^{I,w}$ of the world w of the model.

For each tile $t_i \in T$ we introduce a concept name T_i. Its intended meaning is as follows: we will say that t_k covers an element (i, j) in the grid iff $B_{ij} \subseteq T_k$ (i.e., $B_{ij} \to T_k = \top$). The problem now is to guarantee that every element of the grid is covered by precisely one tile and that the colours of adjacent tiles match *without using too many formulae*. To this end we require $2n$ new concept names $Q_0, \ldots, Q_{2n-1}$; they will encode 2^{2n} worlds w_{ij}, $0 \leqslant i, j < 2^n$.

Precisely we will describe a binary tree of depth $2n$, using $2n$ concept names $Q_0, \ldots, Q_{2n-1}$. This will provide us with 2^{2n} nodes (on the level $2n$) each encoding a world w_{ij}, $0 \leqslant i, j < 2^n$.

Let F_2 be the set of the following formulae (that are similar to those of [**11**], p. 354 and [**14**]):

$$\Box^i\Diamond(Q_i = \top), \Box^i\Diamond(Q_i = \bot), \text{ for } i = 0, \ldots, 2n-1$$

$$\Box^i(Q_j = \Box Q_j), \Box^i(Q_j = \Diamond Q_j), \text{ for } i = 1, \ldots, 2n; j = 0, \ldots, i-1,$$

$$\Box^{2n}\Diamond(Q_i = \Diamond A_i), \Box^{2n+2}(A_i = \top), \text{ for } i = 0, \ldots, 2n-1 \text{ (see Proposition 3.1).}$$

Let F_3 be the set of formulae

$$C_i = \Box^{2n}C_i, C_i = \Diamond^{2n}C_i, \text{ for } i = 0, \ldots, 2n-1,$$

$$T_i = \Box^{2n}T_i, T_i = \Diamond^{2n}T_i, \text{ for } i = 1, \ldots, s.$$

The meaning of the set F_3 is that each C_i (T_j) contains the same objects of domain $\triangle^{I,w}$ (w is root of tree) in every world (on the level $2n$) w_{ij} (i.e., $C_k^{I,w} \subseteq C_k^{I,w_{ij}}$).

Let B, B^r, B^u are three other concept names. B will coincide with B_{ij}, B^r with $B_{i,j+1}$, and B^u with $B_{i+1,j}$ in the world w_{ij} determined by the condition $w_{ij} \models Q_0^{d_0} \wedge \cdots \wedge Q_{2n-1}^{d_{2n-1}} = \top$, where $(d_{2n-1}, \ldots, d_n)$ and $(d_{n-1}, \ldots, d_0)$ are binary representations of i and j, respectively. This will be ensured by the set of formulae F_4:

$$\Diamond^{2n}B = \top, \Box^{2n}\left(B = \bigwedge_{i=0}^{2n-1}\big((C_i \wedge Q_i) \vee (\neg C_i \wedge \neg Q_i)\big)\right).$$

$$\Box^{2n}\left(B^r = \bigvee_{k=0}^{n-1}\left(\neg Q_k \wedge \bigwedge_{j=0}^{k-1} Q_j \wedge \bigwedge_{i=0}^{k-1} \neg C_i \wedge C_k \wedge \bigwedge_{i=k+1}^{2n-1}\big((C_i \wedge Q_i) \vee (\neg C_i \wedge \neg Q_i)\big)\right)\right),$$

$$\Box^{2n}\left(B^u = \bigvee_{k=n}^{2n-1}\left(\neg Q_k \wedge \bigwedge_{j=n}^{k-1} Q_j \wedge \bigwedge_{i=n}^{k-1} \neg C_i \wedge C_k \wedge \bigwedge_{i \notin \{n,\ldots,k\}}\big((C_i \wedge Q_i) \vee (\neg C_i \wedge \neg Q_i)\big)\right)\right).$$

Let F_5 be the set of formulae

$$\Box^{2n}(F_i \wedge F_j = \bot), \quad \text{for } i \neq j,$$

$$\Box^{2n}(F_i = \Diamond F_i), \quad \Box^{2n}(F_i = \Box F_i), \quad \text{for } i = 1, \ldots, s,$$

$$\Box^{2n}\Diamond(F_i = \Diamond E_i), \quad \Box^{2n+2}(E_i = \top), \quad \text{for } i = 1, \ldots, s \text{ (see Proposition 3.1).}$$

This means that $\exists_{\leqslant 1} i \in \{1, \ldots, s\} F_i = \top$ and $F_j = \bot$ for all $j \neq i$.

Let F_6 be the set of formulae

$$\Box^{2n}\Big(\bigvee_{j=1}^{s} T_j = \top\Big),$$
$$\Box^{2n}\big(B \wedge T_i = B \wedge F_i\big), \quad \text{for } i = 1, \dots, s.$$

This means that $\exists! i \in \{1, \dots, s\} B \subseteq T_i$ and $B \wedge T_j = \bot$ for all $j \neq i$.

Now we are in a position to write down the set F_7 of formulae which says that the colors of adjacent tiles match:

$$\Box^{2n}\Big(B^r \subseteq \Big(\bigwedge_{i=0}^{n-1} Q_i\Big) \vee \Big(\bigvee_{i=1}^{s} \bigvee_{\text{right}(i)=\text{left}(j)} F_i \wedge T_j\Big)\Big),$$
$$\Box^{2n}\Big(B^u \subseteq \Big(\bigwedge_{i=n}^{2n-1} Q_i\Big) \vee \Big(\bigvee_{i=1}^{s} \bigvee_{\text{up}(i)=\text{down}(j)} F_i \wedge T_j\Big)\Big).$$

We remark that if $B \subseteq T_k$ than $F_k = \top$ and $F_i = \bot$ for all $i \neq k$, so we have

$$B^u \subseteq \Big(\bigvee_{i=1}^{s} \bigvee_{\text{up}(i)=\text{down}(j)} F_i \wedge T_j\Big) \equiv \bigvee_{i=1}^{s}\Big(F_i \wedge \Big(\bigvee_{\text{up}(i)=\text{down}(j)} T_j\Big)\Big) \equiv$$

$$F_k \wedge \Big(\bigvee_{\text{up}(k)=\text{down}(j)} T_j\Big) \equiv \bigvee_{\text{up}(k)=\text{down}(j)} T_j.$$

One can show that $F = F_1 \bigcup \cdots \bigcup F_7$ is as required. □

COROLLARY 3.1. *The satisfaction problem for ALC_M-formulae in the class K,with the expanding domain assumption, is* NEXPTIME-*hard.*

COROLLARY 3.2. *Testing satisfiability of a finite set $\{F_1, \dots, F_n\}$ of ALC_{M^B}-formulas in a class K with expanding domains is* NEXPTIME-*complete.*

PROOF. The upper bound follows from [**2**] and the lower from Theorem 3.1. □

Also, from the Theorem 3.1 follows that the satisfaction problem for ALC_M-formulae in each of the classes K, N, GL, $S4$, and $K4$, with the expanding domain assumption, is NEXPTIME-hard.

THEOREM 3.2. *The satisfaction problem for ALC_M- and ALC_{M^B}-formulae in the class K is* NEXPTIME-*complete (no matter whether the models have constant or expanding domains).*

PROOF. The upper bound follows from [**2**] and [**17**] and the lower bound from the Theorem 3.1 (Corollary 3.1). □

4. Complexity in the class $S5$

In the proof of the Theorem 3.1 we have used only one assertional axiom of the form $(a : C)$. The usage of $(a : C)$ enabled us to claim that at least one of the concepts is not empty. All other formulae were terminological axioms.

Now we will consider satisfaction problem for ALC_{M^B}-formulae in the class $S5$, supposing that we have at most one assertional axiom of the form $(a : C)$ (let us call them $ALC_{M^{B-}}$-formulae).

Using rules $\Box(C_1 = \top) \wedge \Box(C_2 = \top) \equiv \Box(C_1 \wedge C_2 = \top)$, $\Box\Box\varphi \equiv \Box\varphi$, $\Diamond\Box\varphi \equiv \Box\varphi$, $\Box\Diamond\varphi \equiv \Diamond\varphi$ and $\Diamond\Diamond\varphi \equiv \Diamond\varphi$, every set of formulae can be transformed into the equivalent set of formulae of the following form:

$$\Box(C_0 = \top),\ (C_1 = \top),\ (a : C_1^{'}),\ \Diamond(C_2 = \top),\ \ldots,\ \Diamond(C_s = \top).$$

Now we define s-quasimodel (simple quasimodel) for finite sets of formulae. We fix a finite set F of ALC_{M^B}-formulas in the class $S5$ such that

$$F = \{\Box(C_0 = \top), (C_1 = \top), (a : C_1^{'}), \Diamond(C_2 = \top), \ldots, \Diamond(C_s = \top)\}$$

i.e., ALC_M formula

$$\varphi_F = \Box(C_0 = \top) \wedge (C_1 = \top) \wedge (a : C_1^{'}) \wedge \Diamond(C_2 = \top) \wedge \cdots \wedge \Diamond(C_s = \top).$$

With $\operatorname{con}\varphi_F$ we denote the closure under negation of the set of all concepts in φ_F. Without loss of generality we may identify C and $\neg\neg C$, for every concept C; so the set $\operatorname{con}\varphi_F$ is finite and $|\operatorname{con}\varphi_F| < 2\|\varphi_F\|$, where $\|\varphi_F\|$ is the number of symbols in the formula φ_F. We also suppose that $\Diamond D_1$, $\Diamond D_2$,…, $\Diamond D_m$ are all concepts from $\operatorname{con}\varphi_F$ of the form $\Diamond C$.

Definition 4.1. A concept type t for φ_F is a subset of $\operatorname{con}\varphi_F$ such that

1) $C \wedge D \in t$ iff $C, D \in t$, for every $C \wedge D \in \operatorname{con}\varphi_F$,
2) $\neg C \in t$ iff $C \notin t$, for every $C \in \operatorname{con}\varphi_F$.

Let $\mathfrak{T}_F$ be a set of all concept types for φ_F. For $t \in \mathfrak{T}_F$ we will denote $t_{|R} = \{C \in \operatorname{con}\varphi_F \mid \forall R.C \in t\}$ and $t_{|\Diamond} = \{D \in \operatorname{con}\varphi_F \mid \Diamond D \in t\}$.

Definition 4.2. A set of s-quasiworld for φ_F is a set $T = \{T_0, T_1, \ldots, T_s\}$ such that

3) $T_i \subset \mathfrak{T}_F$, for every $i \in \{0, 1, \cdots, s\}$,
4) $T_i \neq \emptyset$, for every $i \in \{0, 1, \cdots, s\}$,
5) $(\forall t \in T_i)C_i \in t$, for every $i \in \{0, 1, \cdots, s\}$,
6) $(\forall t \in T_i)(\forall(\exists R.C) \in \operatorname{con}\varphi_F)(\exists R.C \in t \text{ iff } (\exists t' \in T_i)t_{|R} \subset t' \wedge C \in t')$, for every $i \in \{0, 1, \cdots, s\}$,
7) $(\exists t = t_a \in T_1)C_1^{'} \in t$, for $(a : C_1^{'}) \in F$,

Definition 4.3. A run in $T = \{T_0, T_1, \ldots, T_s\}$ is a function
$r : \{-m, \ldots, -1, 0, 1, \ldots, s\} \to \bigcup_{i=1}^{s} T_i$ such that

8) $r(i) \in T_i$, for every $i \in \{0, 1, \ldots, s\}$ and $r(-k) \in T_0$, for every $k \in \{1, \ldots, m\}$,
9) $r(i)_{|\Diamond} = r(j)_{|\Diamond}$, for every $i, j \in \{-m, \ldots, 1, 0, 1, \ldots, s\}$

10) For every $k \in \{1, \cdots, m\}$, if $D_k \in r(0)_{|\Diamond}$, then $D_k \in r(-k)$.

Definition 4.4. Let $T = \{T_0, \ldots, T_s\}$ be a set of s-quasiworld for φ_F and ρ set of run in it. The pair (T, ρ) is called a s-quasimodel for φ_F if the following holds:

11) $(\forall T_i \in T)(\forall t \in T_i)(\exists r \in \rho)r(i) = t$.

Lemma 4.1. *If the set of formulae F is satisfiable, then there exists s-quasimodel for φ_F.*

Proof. If the set of formulae F is satisfiable, then there exists at least one model for F. Let us consider arbitrary non-empty family of models $MC = \{\mathfrak{M}_c = (W_c, W_c \times W_c, I_c) \mid \mathfrak{M}_c \models F \text{ and } c \in C\}$, where $C = |MC| \geqslant 1$. We will construct s-quasimodel for φ_F which corresponds to the family MC.

For every model $\mathfrak{M}_c = (W_c, W_c \times W_c, I_c) \in MC$, for every world $w \in W_c$ and for every object $x \in \Delta^{I_c,w}$, let us define the concept type $t^{I_c,w}(x) = \{C \in \operatorname{con}\varphi_F \mid x \in C^{I_c,w}\}$.

Let $T_i = \bigcup_{c \in C}\{t^{I_c,w}(x) \mid w \in W_c, w \models (C_i = \top), x \in \Delta^{I_c,w}\}$. For arbitrary $t \in T_i, i = 0, \ldots, s$, let us construct the run $r = r_t$ such that $r(i) = t$. It follows that, for every $t \in T_i$, there exists a model $\mathfrak{M} = (W, W \times W, I) \in MC$ and there exists a world $w \in W$ and there exists an objects $x \in \Delta^{I,w}$, such that $w \models (C_i = \top) \wedge t = t^{I,w}(x)$. Since $\mathfrak{M} \models F$, it follows that $(\forall j \in \{0, \ldots, s\})(\exists w_j \in W)w_j \models (C_j = \top)$. Let $t_j = t^{I,w_j}(x)$. Obviously $t_j \in T_j$. If $\Diamond D_k \notin t$, put $t_{-k} = t_0$. Now suppose that $\Diamond D_k \in t$. Since $\Diamond D_k \in t$ iff $\Diamond D_k \in t^{I,w}(x)$ iff $x \in (\Diamond D_k)^{I,w}$ iff $(\exists v_k \in W)x \in (D_k)^{I,v_k}$, we can define $t_{-k} = t^{I,v_k}(x)$. Since $v_k \models (C_0 = \top)$, it follows that $t_{-k} \in T_0$. Now, we put $r(j) = t_j$ for every $j \in \{-m, \ldots, s\}$. Let ρ be the set of all runs constructed in that way. Then, by construction, the tuple $(\{T_0, T_1, \ldots, T_s\}, \rho)$ is a s-quasimodel for φ_F. □

Lemma 4.2. *If there exists a s-quasimodel for φ_F, then the set of formulae F is satisfiable.*

Proof. Let $Q = (\{T_0, T_1, \ldots, T_s\}, \rho)$ be a s-quasimodel for φ_F. Based on Q, we will construct the model for F as follows. Let $\rho' = \{r' = r'(r,k) \mid r \in \rho, k \in \{0, \ldots, m\}\ \}$, where

$r'(r,0) = r$,

$r'(r,k)(i) = r'(i) = r(i)$ for $k \neq 0$ and $i \notin \{0, -k\}$,

$r'(r,k)(0) = r'(0) = r(-k)$ for $k \neq 0$ and $i = 0$,

$r'(r,k)(-k) = r'(-k) = r(0)$ for $k \neq 0$ and $i = -k$.

From the set of all runs ρ' that goes through t_a we extract one run which we denote by r_a. Now we construct the model for the set F. Let

$V = \{v_{-m}, \ldots, v_{-1}, v_0, v_1, \ldots, v_s\}$ be the set of worlds,

let $\lhd = V \times V$ be the relation,

let $I(v_j) = \left\langle \Delta^{I,v_j}, R_0^{I,v_j}, \ldots, A_0^{I,v_j}, \ldots, a^{I,v_j} \right\rangle$ be the interpretation, where $\Delta^{I,w} = \Delta^I = \{r \mid r \in \rho'\}$, $A^{I,v_j} = \{r \in \Delta^I \mid A \in r(j)\}$ (for atomic concepts A), $R^{I,v_j} = \{(r', r'') \in (\Delta^I)^2 \mid r'(j)_{|R} \subseteq r''(j)\}$ and $a^{I,v_j} = r_a$, so that the model for the set F is $\mathfrak{M} = (V, \lhd, I)$.

It still remains to prove, by the induction on concept complexity, that $C \in r(i)$ iff $r \in C^{I,v_i}$. The most important cases are $C \equiv \Diamond D_k$ and $C \equiv (\exists R.D)$:

In the first case we have: $\Diamond D_k \in r(i)$ iff $D_k \in r(-k)$ (if $i = -k$ take simetrically $r'(i)$) iff $r \in D_k^{I,v_{-k}}$ iff $(\exists v_{-k})(v_i \lhd v_{-k})(r \in D_k^{I,v_{-k}})$ iff $r \in (\Diamond D_k)^{I,v_i}$

In the second case we have: $(\exists R.D) \in r(i)$ iff $(\exists t' \in T_i) r(i)_{|R} \subseteq t' \wedge D \in t'$ (notice that $(\exists r' \in \rho) r'(i) = t'$) iff $(\exists r'(i) \in T_i) r(i)_{|R} \subseteq r'(i) \wedge D \in r'(i)$ iff $(\exists r' \in \Delta^I)(r, r') \in R^{I,v_i} \wedge r' \in D^{I,v_i}$ iff $r \in (\exists R.D)^{I,v_i}$

Finally we notice that, for arbitrary $i \in \{0, \ldots, s\}$, we have $(\forall t \in T_i) C_i \in t$ iff $(\forall r \in \rho') C_i \in r(i) \in T_i$ iff $r \in C_i^{I,v_i}$ iff $C_i^{I,v_i} = \Delta^I$ iff $v_i \models (C_i = \top)$. It is now obvious that in the model $\mathfrak{M}$ we have $v_1 \models F$. □

We now give the algorithm of satisfiability: Starting from the set F, construct the sets con φ_F and $\mathfrak{T}_F$. Let $T_0 = \{t \in \mathfrak{T}_F \mid C_0 \in t\}$ and $T_i = \{t \in T_0 \mid C_i \in t\}$, $i = 1, \ldots, s$.

Repeat steps (a)-(c) as many times as it is possible:

(a) If there exists $T_i = \emptyset$ or $C_1' \notin t$ for all $t \in T_1$, then the algorithm returns the answer "NO".
(b) If there exists $t \in T_i$, for which condition 6) of the definition 4.2 fails, then exclude t from T_i.
(c) If $t \in T_i$ is such that we can not construct a run through it, then exclude t from T_i.

If none of the rules (a)-(c) can be applied, then the algorithm returns the answer "YES".

Lemma 4.3. *If the algorithm for the set F returns "YES", then there exists a s-quasimodel for φ_F i.e. the set of formulae F is satisfiable.*

Proof. The algorithm constructs set $T = \{T_0, T_1, \ldots, T_s\}$. The conditions 3) and 5) of definition 4.2 are fulfilled by construction. Since the rule (a) of the algorithm was not applied, the conditions 4) and 7) of definition 4.2 are met, and since we can not apply the rule (b) of the algorithm, it follows that condition 6) of definition 4.2 is fulfilled. Hence, T is set of s-quasiworld.

For each $t \in T_i$, there exists a run $r = r_t$ which goes through it, since otherwise, we would be able to apply rule (c) of algorithm. If ρ is the set of all runs, the pair (T, ρ) is s-quasimodel for φ_F. □

Lemma 4.4. *If the algorithm for the set F returns "NO", then s-quasimodel for φ_F does not exist, so the set of formulae F is not satisfiable.*

Proof. Assume the opposite, let (T', ρ') be s-quasimodel for φ_F, where $T' = \{T_0', T_1', \ldots, T_s'\}$. In the very beginning, the algorithm constructs sets $T_0, T_1, \ldots, T_s$ such that $T_i' \subseteq T_i$. Applying the rules (b) and (c) of the algorithm, we can not exclude elements from T_i which belong to T_i', so that rule (a) can never be applied. □

Corollary 4.1. *The algorithm is correct.*

Lemma 4.5. *The algorithm stops in at most exponential number of steps, according to the size of the input.*

Proof. Since $|\mathfrak{T}_F| \leqslant 2^{|\operatorname{con}\varphi_F|}$ and $T_i \subset T$, for $i = 0, 1, \dots, s$, it is clear that the construction of $\mathfrak{T}_F$ and T_i requires EXPTIME. Hence, we have at most exponential number of concept types to which rules (a)-(c) can be applied. Also, after each step, there is one type less, so that the rules (b) and (c) of the algorithm can be applied at most exponential number of times. To check whether we can apply some rule, we need at most exponential time. Hence, the algorithm will return the answer after at most exponential number of steps. □

Note that the satisfaction problem for ALC_M-formulae without modal operators is EXPTIME-hard (see e.g [**13**]), so that we have the following theorem.

Theorem 4.1. *The satisfaction problem for $ALC_{M^{B^-}}$-formulae in the class $S5$ is* EXPTIME-*complete*

Lemma 4.6. *The satisfaction problem for ALC_M-formulae in the class $S5$ is* NEXPTIME-*complete*

Proof. See [**14**] and [**13**]. □

If the simulation of disjunction between formulae in description logics with modal operators based on the class $S5$ had been possible, then the complexity of satisfaction problems in ALC_M and $ALC_{M^{B^-}}$-formulae would have been the same. These means that, assuming P $\neq$ NP (i.e. EXPTIME $\neq$ NEXPTIME), the Theorem 4.1 and the Lemma 4.6 tells us that the simulation of disjunction is not possible in $ALC_{M^{B^-}}$.

References

[1] F. Baader and B. Hollunder, *A terminological knowledge representation system with complete inference algorithms*, in: *Proceedings of the workshop on Processing Declarative Knowledge, PDK-91*, Springer Verlag, 1991, 67–86.

[2] F. Baader and A. Laux, *Terminological logics with modal operators*, in: *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, Morgan Kaufman, Montreal, Canada, 1995, 808–814.

[3] F. Baader and H. Ohlbach, *A multi-dimensional terminological knowledge representation language*, in: *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, 1993, 690–695.

[4] van Emde Boas, *The convenience of tilings*, Tehnical Report CT-96-01, Institute for Logic, Language and Computation, University of Amsterdam.

[5] A. Borgida, R. J. Brachman, D. L. McGuiness, and L. A. Resnick, *CLASSIC: A structural data model for objects*, in: *Proceedings of the ACM SIGMOD International Conference on Management of Data*, Portland, Oreg., 1989, 59–67.

[6] R. J. Brachman and J. G. Schmolze, *An overview of the KL-ONE knowledge representation system*, Cognitive Science **9** (1985), 171–216.

[7] F. Donini, M. Lenzerini, D. Nardi, and W. Nutt, *The complexity of concept languages*, Technical Report RR-95-07, Deutsches Forchungszentrum für Künstliche Intelligenz (DFKI), 1995.

[8] F. Donini, M. Lenzerini, D. Nardi, and A. Schaerf, *Reasoning in description logics*, in: G. Brewka, *Principles of Knowledge Representation*, CSLI Publications, 1996, 191–236.

[9] G. De Giacomo, *Decidability of Class-Based Knowledge Representation Formalisms*, PhD thesis, Univ. di Roma, 1995.
[10] G. De Giacomo and M. Lenzerini, *TBox and ABox reasoning in expressive description logics*, in: *Proceedings of the fifth Conference on Principles of Knowledge Representation and Reasoning*,Morgan Kaufman, Montreal, Canada, 1996.
[11] J. Halpern and Yo. Moses. *A guide to completeness and complexity for modal logics of knowledge and belief*, Artificial Intelligence **54** (1992),319–379.
[12] A. Laux. *Beliefs in multi-agent worlds: a terminological approach*, in: *Proceedings of the 11th European Conference on Artificial Intelligence*, Amsterdam, 1994, 299–303.
[13] M. Mosurović. *Složenost opisnih logika s modalnim operatorima*, Doktorska disertacija, Univerzitet u Beogradu, 2000.
[14] M. Mosurović and M. Zakharyaschev, *On the complexity of description logics with modal operators*, in: P. Kolaitos and G. Koletos, *Proceedings of the 2nd Panhellenic Logic Symposium*, Delphi, 1999, 166–171.
[15] K. Schild, *Combining terminological logics with tense logic*, in: *Proceedings of the 6th Portuguese Conference on Artificial Intelligence*, Porto, 1993, 105–120.
[16] F. Wolter and M. Zakharyaschev, *Dynamic description logic*, Advances in Modal Logic. Volume II, CSLI Publications, Stanford, 1999.
[17] F. Wolter and M. Zakharyaschev, *Satisfiability problem in description logics with modal operators*, in: *Proceedings of the sixth Conference on Principles of Knowledge Representation and Reasoning*, Morgan Kaufman, Montreal, Canada, 1998, 512–523.
[18] F. Wolter and M. Zakharyaschev, *Temporalizing description logics*, Frontiers of Combining Systems, Kluwer Academic Publishers, 1999.
[19] F. Wolter and M. Zakharyaschev, *Multi-dimensional description logics*, in: *Proceedings of IJCAI'99*, Stockholm, 1999.

Prirodno-matematički fakultet
81000 Podgorica
Montenegro
milenko@cg.ac.yu