Fundamenta Informaticae XX (2006) 1–25
IOS Press

Complexity issues in multiagent logics

Marcin Dziubiński
Department of Economics
Lancaster University
LA1 4YX Lancaster, UK
amosild@tlen.pl

Barbara Dunin-Kęplicz
Institute of Informatics
Warsaw University
Banacha 2
02-097 Warsaw, Poland
keplicz@mimuw.edu.pl

Institute of Computer Science
Polish Academy of Sciences
Ordona 21
01-237 Warsaw, Poland

NIAS
Meijboomlaan 1
2242 PR Wassenaar
The Netherlands

Rineke Verbrugge
Institute of Artificial Intelligence
University of Groningen
Grote Kruisstraat 2/1
9712 TS Groningen
The Netherlands
rineke@ai.rug.nl

NIAS
Meijboomlaan 1
2242 PR Wassenaar
The Netherlands

Abstract. Our previous research presents a methodology of cooperative problem solving for belief-desire-intention (BDI) systems, based on a complete formal theory called TEAMLOG. This covers both a static part, defining individual, bilateral and collective agent attitudes, and a dynamic part, describing system reconfiguration in a dynamic, unpredictable environment. In this paper, we investigate the complexity of the satisfiability problem of the static part of TEAMLOG, focusing on individual and collective attitudes up to collective intention. Our logics for teamwork are squarely multi-modal, in the sense that different operators are combined and may interfere. One might expect that such a combination is much more complex than the basic multi-agent logic with one operator, but in fact we show that it is not the case: the individual part of TEAMLOG is PSPACE-complete, just like the single modality case. The full system, modelling a subtle interplay between individual and group attitudes, turns out to be EXPTIME-complete, and remains so even when propositional dynamic logic is added to it.

Additionally we make a first step towards restricting the language of TEAMLOG in order to reduce its computational complexity. We study formulas with bounded modal depth and show that in case of the individual part of our logics, we obtain a reduction of the complexity to NPTIME-completeness. We also show that for group attitudes in TEAMLOG the satisfiability problem remains in EXPTIME-hard, even when modal depth is bounded by 2. We also study the combination of reducing modal depth and the number of propositional atoms. We show that in both cases this allows for checking the satisfiability in linear time.

1. Introduction

In this paper, we investigate the complexity of two important subsystems of the teamwork logics constructed (TEAMLOG) in our previous papers ([9, 10, 11]). Although the results and methods of this paper may be applied to a complexity analysis of many multi-modal logics combining different but interrelated agent attitudes, the jumping point of the paper is our own theory of teamwork as presented in [9]. Let us give a reminder of this theory, as well as on complexity theory as it is relevant for logics applied in multiagent systems (MAS).

1.1. TEAMLOG: A Formal Theory of Mental States in Teamwork

When constructing belief-desire-intention (BDI) systems, the first research question has been to create a model of an agent as an *individual, autonomous* entity. A more recent goal has been to organize agents' cooperation in a way allowing the achievement of their, possibly complex, common goal, while preserving, at least partly, the autonomy of particular agents involved. The BDI-model naturally comprises such individual notions like *beliefs*, referring to the agent's informational attitudes, as well as *goals* and *intentions*, dealing with its motivational stance. However in teamwork, when a team of agents needs to work together in a planned and coherent way, these are not enough: the group as a whole needs to present a common collective attitude over and above individual attitudes of team members. Without this, a sensible organization of cooperation seems to be impossible: the existence of collective (or joint) motivational attitudes is a necessary condition for a group of agents to become a cooperative team [9, 11]. Thus, in the context of teamwork agents' attitudes are considered on the individual, social (i.e. bilateral) and collective level.

A theory of informational attitudes on the group level has been formalized in terms of epistemic logic [14]. As regards motivational attitudes, the situation is much more complex: a conceptually coherent theory was vitally needed in the MAS literature, as the notions on the bilateral and collective level cannot be viewed as a straightforward extension or a sort of sum of individual notions. In order to define them, many subtle aspects that are hard to formalize need to be introduced. A departure point to construct a *static, descriptive* theory of collective motivational attitudes is formed by individual goals, beliefs and intentions of cooperating agents. Research on teamwork should address the question what it means for a group of agents to have a *collective intention*, and then a *collective commitment* to achieve a common goal.

In our approach, the fundamental role of collective intention is to consolidate a group as a cooperating team, while collective commitment leads to team action, i.e., to coordinated realization of individual actions by the agents that have committed to do them according to the team plan. Both notions are constructed in a way that allows us to fully express the potential of strictly cooperative teams [9, 11]. In this paper we will focus on the theory of teamwork, TEAMLOG, built up from the above-mentioned individual and collective attitudes up to collective intentions, where our goal is to give the exact complexity class of the full logic and its most important subsystem for individual attitudes.

When modelling collective intention, agents' awareness about the overall situation needs to be taken into account. In a theory of multiagent systems, the essential notion of awareness is understood as a reduction of the general sense of this notion to the state of an agent's beliefs about itself, about other agents and about the state of the environment. Under this assumption, various epistemic logics and various notions of group information (from distributed belief to common knowledge) are adequate to formalize agents' awareness [14, 11]. In the presented theory, group awareness is usually expressed in terms of the (rather strong) notion of *common belief*, but one can also consider weaker forms depending on the circumstances in question. In the complexity analysis, we do *not* take into account the cognitive and other processes necessary for establishing attitudes that appear in the definitions found in section 2; for such an approach, see [12]. We are just interested in showing how complex it is to check *satisfiability* and validity of the formulas with respect to TEAMLOG. Let us turn to a short reminder about the decidability and complexity of such important questions about logical theories.

1.2. Computational complexity

In this paper we investigate the complexity of two particular modal logics for multiagent systems. In particular, we examine the complexity of their *satisfiability problem*: given a formula φ, how much time and space (in terms of the length of φ) are needed to compute whether φ is satisfiable, i.e. whether there is a suitable Kripke model $\mathcal{M}$ (from the class of structures corresponding to the logic) and a world s in it, such that $\mathcal{M}, s \models \varphi$? From this, the complexity of the *validity problem* (truth in all worlds in all suitable Kripke models) follows immediately, because φ is valid if and only if $\neg\varphi$ is not satisfiable. *Model checking*, i.e. evaluating truth of a given formula in a given world and model ($\mathcal{M}, s \models \varphi$) is the most important related problem, and is easily seen to be less complex than both satisfiability and validity. Thus, for example, if some logic' satisfiability problem is NP-complete, then its validity problem is coNP-complete. We do not investigate the complexity of model checking here, but see [19] for such an analysis of some MAS logics; in any case, various methods have already been developed that can perform model checking in a reasonable time, as long as the considered models are not too large.

Unfortunately, even satisfiability for propositional logic is an NP-complete problem. Thus, if indeed $P \neq NP$, then modal logics interesting for MAS, all containing propositional logic as a subsystem, do not have efficiently solvable satisfiability problems. Even though a single efficient algorithm performing well on all inputs is not possible, it is still important to discover in which complexity class a given logical theory falls. In our work we take the point of view of the system developer who wants to reason about, specify and verify a multiagent system to be constructed. It turns out that for many of the interesting formulas appearing in such human reasoning, satisfiability tends to be easier to compute than suggested by the worst-case labels like "PSPACE-complete" and "EXPTIME-complete" [16]. It would be helpful to develop automated, efficient tools to support the system developer with some reasoning tasks, and in the discussion (section 5) we will come back to methods that can simplify satisfiability problems for MAS logics in an application-dependent way.

Of many single-agent modal logics with one modality, the complexity has long been known. An overview is given in [16], which extends these results to multi-agent logics, though still containing only a single modality (either knowledge or belief). For us, the following results are relevant. The satisfiability problems for the systems $\mathrm{S5}_1$ and $\mathrm{KD45}_1$, modelling knowledge and belief of one agent, are NP-complete. Thus, perhaps surprisingly, they are no more complex than propositional logic. The complexity is increased to PSPACE if these systems are extended to more than one agent. PSPACE is also the complexity class of satisfiability for many other modal logics, for both the single and the multiagent case; examples are the basic system K_n (that we adopt for goals) and the system KD_n (that we adopt for intentions, see section 2). As soon as a notion of seemingly infinite character such as common knowledge or common belief (everybody believes and everybody believes that everybody believes and ...) is modelled, the complexity of the satisfiability problem jumps to EXPTIME. Intuitively, trying to find a satisfying model for a formula containing a common belief operator by the tableau method, one may need to look exponentially deep in the tableau tree to find it, while for simpler modal logics like K_n, a depth-first search through a polynomially shallow tree suffices for all formulas.

When investigating the complexity of multi-modal logics, one might like to turn to general results on the transfer of the complexity of satisfiability problems from single logics to their combinations: isn't a combination of a few PSPACE-complete logics, with some simple interdependency axioms, automatically PSPACE-complete again? However, it turns out that the positive general results that do exist (such as those in [4]) apply mainly to minimal combinations, without added interdependencies, of two NP-complete systems, each with a single modality. Even more dangerously, there are some very negative results on the transfer of complexity to combined systems. Thus, there are two "very decidable" logics whose combination, even without any interrelation axioms, is undecidable[1]. This goes to show that one needs to be very careful with any assumptions about generalizations of complexity results to combined systems.

Our logic TEAMLOG and its subsystems are squarely multi-modal, not only in the sense of modelling a multi-agent version of one modal operator, but also in the sense that different operators are combined and may interfere. One might expect that such a combination is much more complex than the basic multiagent logic with one operator, but in fact we show in this paper that this is not the case: the "individual part" of TEAMLOG (called TEAMLOG$^{\mathrm{ind}}$) is PSPACE-complete. In order to prove this, the semantic properties relating to the interdependency axioms must be carefully translated to conditions on the multi-modal tableau with which satisfiability is tested. Of course the challenging question appears when informational and motivational group notions are added to this individual part. We show that also for this expressive system, modelling a subtle interplay between individual and group attitudes, satisfiability is EXPTIME-complete, thus of the same complexity as the system only modelling common belief. As a bonus, it turns out that even adding dynamic logic (which is relevant for our study of the evolution of motivational attitudes in changing environments [10]) does not increase complexity beyond EXPTIME.

Finally, inspired by [15], we explore some possibilities of lowering the complexity of the satisfiability problem by restricting the modal depth of the formulas concerned or by limiting the number of propositional atoms used in the language. It turns out that bounding the depth gives a nice reduction in the individual case, but is less successful where group attitudes are concerned. Combining modal depth reduction with bounding the number of propositional atoms allows for checking the satisfiability in linear time.

[1] For B, take a variant of dynamic logic with two atomic programs, both deterministic. Take ; and $\cap$ as only operators. Satisfiability of formulas with respect to B, like that for propositional dynamic logic itself, is in EXPTIME. For C, take the logic of the global operator A (Always), defined as follows: $M, w \models A\varphi$ iff for all $v \in W$, $M, v \models \varphi$. Satisfiability for C is in NP. In [4] (see also [3, Theorem 6.31]), it is shown that the minimal combination of B and C is not only *not* in EXPTIME, but even undecidable in any finite time.

$\mathrm{BEL}(a, \varphi)$	agent a has the belief that φ
$\text{E-BEL}_G(\varphi)$	every agent in group G has the belief that φ
$\text{C-BEL}_G(\varphi)$	group G has the common belief that φ
$\mathrm{GOAL}(a, \varphi)$	agent a has as a goal to achieve φ
$\mathrm{INT}(a, \varphi)$	agent a has the intention to achieve φ
$\text{E-INT}_G(\varphi)$	every agent in group G has the individual intention to achieve φ
$\text{M-INT}_G(\varphi)$	group G has the mutual intention to achieve φ
$\text{C-INT}_G(\varphi)$	group G has the collective intention to achieve φ

Table 1. Formulas and their intended meaning

The rest of the paper is structured as follows. Section 2 shortly reviews the language, semantics and axiom systems for the individual and group parts of our teamwork logics. In section 3, the complexity of the satisfiability problem for TEAMLOG$^{\text{ind}}$, the part of the theory that covers individual attitudes, is investigated. This is done both for the system as a whole and for some restrictions of it where formulas have bounded modal depth or the number of propositional atoms is bounded.

Section 4 extends the investigation to the theory TEAMLOG covering the group notions of common belief and collective intentions. Also here, we study both the theory as a whole and its restriction to formulas of bounded modal depth or of bounded number of propositional atoms. Finally, in section 5 we discuss the results and present some avenues for possible extensions. This paper is an extension of [13].

2. Logical Background

As mentioned before, we propose the use of multi-modal logics to formalize agents' informational and motivational attitudes as well as actions they perform. In the present paper, where we restrict ourselves to the static aspects of the agents' mental states, we only present axioms relating attitudes of agents with respect to *propositions*, not actions. A proposition reflects a particular state of affairs.

Table 1 gives the formulas appearing in this paper, together with their intended meanings. The symbol φ denotes a proposition.

2.1. The Language

Formulas are defined with respect to a fixed finite set of agents. The basis of the inductive definition is given in the following definition.

Definition 2.1. (Language)
The language is based on the following two sets:

- a countable set $\mathcal{P}$ of *propositional symbols*;
- a finite set $\mathcal{A}$ of *agents*, denoted by numerals $1, 2, \ldots, n$.

Definition 2.2. (Formulas)
We inductively define the set $\mathcal{L}$ of formulas as follows.

F1 each atomic proposition $p \in \mathcal{P}$ is a formula;

F2 if φ and ψ are formulas, then so are $\neg\varphi$ and $\varphi \wedge \psi$;

F4 if φ is a formula, $i \in \mathcal{A}$, and $G \subseteq \mathcal{A}$, then the following are formulas:

epistemic modalities $\mathrm{BEL}(i, \varphi)$, $\text{E-BEL}_G(\varphi)$, $\text{C-BEL}_G(\varphi)$;

motivational modalities $\mathrm{GOAL}(i, \varphi)$, $\mathrm{INT}(i, \varphi)$, $\text{E-INT}_G(\varphi)$, $\text{M-INT}_G(\varphi)$, $\text{C-INT}_G(\varphi)$.

The standard propositional constants and connectives $\top$, $\bot$, $\vee$, $\rightarrow$ and $\leftrightarrow$ are defined in the usual way.

2.2. Semantics Based on Kripke Models

Each Kripke model for the language $\mathcal{L}$ consists of a set of worlds, a set of accessibility relations between worlds, and a valuation of the propositional atoms, as follows.

Definition 2.3. (Kripke model)
A Kripke model is a tuple
$\mathcal{M} = (W, \{B_i : i \in \mathcal{A}\}, \{G_i : i \in \mathcal{A}\}, \{I_i : i \in \mathcal{A}\}, \mathit{Val})$, such that

1. W is a set of possible worlds, or states;

2. For all $i \in \mathcal{A}$, it holds that $B_i, G_i, I_i \subseteq W \times W$. They stand for the accessibility relations for each agent with respect to *beliefs*, *goals*, and *intentions*, respectively. For example, $(s, t) \in B_i$ means that t is an epistemic alternative for agent i in state s.

3. $\mathit{Val} : \mathcal{P} \times W \to \{0, 1\}$ is a valuation function that assigns the truth values to atomic propositions in states.

A Kripke frame $\mathcal{F}$ is defined as a Kripke model, but without the valuation function. At this stage, it is possible to define the truth conditions pertaining to the language $\mathcal{L}$. The expression $\mathcal{M}, s \models \varphi$ is read as "formula φ is satisfied by world s in structure $\mathcal{M}$".

Define world t to be G_B*-reachable* (respectively G_I*-reachable*) from world s iff $(s, t) \in (\bigcup_{i \in G} B_i)^+$ (respectively $(s, t) \in (\bigcup_{i \in G} I_i)^+$). Formulated more informally, this means that there is a path of length ≥ 1 in the Kripke model from s to t along accessibility arrows B_i (respectively I_i) that are associated with members i of G.

Definition 2.4. (Truth definition)

- $\mathcal{M}, s \models p$ iff $\mathit{Val}(p, s) = 1$;

- $\mathcal{M}, s \models \neg\varphi$ iff $\mathcal{M}, s \not\models \varphi$;

- $\mathcal{M}, s \models \varphi \wedge \psi$ iff $\mathcal{M}, s \models \varphi$ and $\mathcal{M}, s \models \psi$;

- $\mathcal{M}, s \models \mathrm{BEL}(i, \varphi)$ iff $\mathcal{M}, t \models \varphi$ for all t such that sB_it;

- $\mathcal{M}, s \models \mathrm{GOAL}(i, \varphi)$ iff $\mathcal{M}, t \models \varphi$ for all t such that sG_it;

- $\mathcal{M}, s \models \mathrm{INT}(i, \varphi)$ iff $\mathcal{M}, t \models \varphi$ for all t such that sI_it;

- $\mathcal{M}, s \models \text{E-BEL}_G(\varphi)$ iff for all $i \in G$, $\mathcal{M}, s \models \text{BEL}(i, \varphi)$;
- $\mathcal{M}, s \models \text{C-BEL}_G(\varphi)$ iff $\mathcal{M}, t \models \varphi$ for all t that are G_B-reachable from s.
- $\mathcal{M}, s \models \text{E-INT}_G(\varphi)$ iff for all $i \in G$, $\mathcal{M}, s \models \text{INT}(i, \varphi)$;
- $\mathcal{M}, s \models \text{M-INT}_G(\varphi)$ iff $\mathcal{M}, t \models \varphi$ for all t that are G_I-reachable from s.

In particular, this implies that for all models $\mathcal{M}$ and states s, $\mathcal{M}, s \models \top$ and $\mathcal{M}, s \not\models \bot$.

2.3. Axiom Systems for Individual and Collective Attitudes

Let us give a reminder of TEAMLOG$^{\text{ind}}$ for individual attitudes and their interdependencies (see subsections 2.3.1, 2.3.2, 2.3.3, 2.3.4), followed by our additional axioms and rules for group attitudes (see subsections 2.3.5, 2.3.6). These axioms and rules, together forming TEAMLOG, are fully explained in [9]. All axiom systems introduced here are based on the finite set $\mathcal{A}$ of n agents.

2.3.1. General Axiom and Rule

The following axiom and rule, covering propositional reasoning, form part and parcel of any system of normal modal logic:

P1 All instances of propositional tautologies;

PR1 From φ and $\varphi \to \psi$, derive ψ; (Modus Ponens)

2.3.2. Axioms and Rules for Individual Belief

The well-known system KD45$_n$ consists of the following for each $i \in \mathcal{A}$:

A2 $\text{BEL}(i, \varphi) \wedge \text{BEL}(i, \varphi \to \psi) \to \text{BEL}(i, \psi)$ (Belief Distribution)

A4 $\text{BEL}(i, \varphi) \to \text{BEL}(i, \text{BEL}(i, \varphi))$ (Positive Introspection)

A5 $\neg\text{BEL}(i, \varphi) \to \text{BEL}(i, \neg\text{BEL}(i, \varphi))$ (Negative Introspection)

A6 $\neg\text{BEL}(i, \bot)$ (Consistency)

R2 From φ infer $\text{BEL}(i, \varphi)$ (Belief Generalization)

2.3.3. Axioms for Individual Motivational Operators

For goals, we take the system K$_n$ and for intentions the system KD$_n$, as follows, for each $i \in \mathcal{A}$:

A2$_D$ $\text{GOAL}(i, \varphi) \wedge \text{GOAL}(i, \varphi \to \psi) \to \text{GOAL}(i, \psi)$ (Goal Distribution)

A2$_I$ $\text{INT}(i, \varphi) \wedge \text{INT}(i, \varphi \to \psi) \to \text{INT}(i, \psi)$ (Intention Distribution)

R2$_D$ From φ infer $\text{GOAL}(i, \varphi)$ (Goal Generalization)

R2$_I$ From φ infer $\text{INT}(i, \varphi)$ (Intention Generalization)

A6$_I$ $\neg\text{INT}(i, \bot)$ for $i = 1, \ldots, n$ (Intention Consistency)

2.3.4. Interdependencies Between Intentions and Other Attitudes

For each $i \in \mathcal{A}$:

A7$_{DB}$ $\mathrm{GOAL}(i,\varphi) \to \mathrm{BEL}(i, \mathrm{GOAL}(i,\varphi))$ (Positive Introspection for Goals)

A7$_{IB}$ $\mathrm{INT}(i,\varphi) \to \mathrm{BEL}(i, \mathrm{INT}(i,\varphi))$ (Positive Introspection for Intentions)

A8$_{DB}$ $\neg\mathrm{GOAL}(i,\varphi) \to \mathrm{BEL}(i, \neg\mathrm{GOAL}(i,\varphi))$ (Negative Introspection for Goals)

A8$_{IB}$ $\neg\mathrm{INT}(i,\varphi) \to \mathrm{BEL}(i, \neg\mathrm{INT}(i,\varphi))$ (Negative Introspection for Intentions)

A9$_{ID}$ $\mathrm{INT}(i,\varphi) \to \mathrm{GOAL}(i,\varphi)$ (Intention implies Goal)

By TeamLog$^{\mathrm{ind}}$ we denote the axiom system consisting of all the above axioms and rules for individual beliefs, goals and intentions as well as their interdependencies.

2.3.5. Axioms and Rule For General ("Everyone") and Common Belief

C1 $\text{E-BEL}_G(\varphi) \leftrightarrow \bigwedge_{i \in G} \mathrm{BEL}(i,\varphi)$ (General Belief)

C2 $\text{C-BEL}_G(\varphi) \leftrightarrow \text{E-BEL}_G(\varphi \wedge \text{C-BEL}_G(\varphi))$ (Common Belief)

RC1 From $\varphi \to \text{E-BEL}_G(\psi \wedge \varphi)$ infer $\varphi \to \text{C-BEL}_G(\psi)$ (Induction Rule)

2.3.6. Axioms and Rule for Mutual and Collective Intentions

M1 $\text{E-INT}_G(\varphi) \leftrightarrow \bigwedge_{i\in G} \mathrm{INT}(i,\varphi)$ (General Intention)

M2 $\text{M-INT}_G(\varphi) \leftrightarrow \text{E-INT}_G(\varphi \wedge \text{M-INT}_G(\varphi))$ (Mutual Intention)

M3 $\text{C-INT}_G(\varphi) \leftrightarrow \text{M-INT}_G(\varphi) \wedge \text{C-BEL}_G(\text{M-INT}_G(\varphi))$ (Collective Intention)

RM1 From $\varphi \to \text{E-INT}_G(\psi \wedge \varphi)$ infer $\varphi \to \text{M-INT}_G(\psi)$ (Induction Rule)

By TeamLog we denote the union of TeamLog$^{\mathrm{ind}}$ with the above axioms and rules for general and common beliefs and for general, mutual and collective intentions.

2.4. Correspondences Between Axiom Systems and Semantics

Most of the axioms above, as far as they do not hold on all frames like **A2**, correspond to well-known structural properties on Kripke frames. Thus, the axiom **A4** holds in a Kripke frame $\mathcal{F}$ iff all B_i relations are transitive; **A5** holds iff all B_i relations are Euclidean; and **A6** holds iff all B_i relations are serial (for proofs of these correspondences and correspondence theory in general, see [2]). As for the interdependencies, the semantic property corresponding to **A7**$_{IB}$ is $\forall s,t,u((sB_it \wedge tI_iu) \to sI_iu)$, analogously for **A7**$_{GB}$. The property that corresponds to **A8**$_{IB}$ is $\forall s,t,u((sI_it \wedge sB_iu) \to uI_it)$, analogously for **A8**$_{GB}$. Finally, for **A9**$_{IG}$ the corresponding semantic property is $G_i \subseteq I_i$. For proofs of these correspondences, see [11].

Note also that the Induction Rules **RC1** and **RM1** are sound due to the definitions of G_B-reachability and G_I-reachability in terms of the transitive closure of the union of individual relations for group G, respectively; see Definition 2.4.

3. Complexity of TEAMLOG$^{\text{ind}}$

We will show that the satisfiability problem for TEAMLOG$^{\text{ind}}$ is PSPACE-complete. First we present an algorithm for deciding satisfiability of a TEAMLOG$^{\text{ind}}$ formula φ working in polynomial space, thus showing that the satisfiability problem is in PSPACE. The construction of the algorithm and related results are based on the method presented in [16]. The method is centred around the well known notions of a *propositional tableau*, a *fully expanded propositional tableau* (a set that along with any formula ψ contained in it, contains also all its subformulas, each of them either in positive or negated form), and a *tableau* designed for a particular system of multimodal logic. Let us give adaptations of the most important definitions from [16] as a reminder:

Definition 3.1. (Propositional tableau)
A *propositional tableau* is a set T of formulas such that:

1. if $\neg\neg\psi \in T$ then $\psi \in T$;
2. if $\varphi \wedge \psi \in T$ then both $\varphi, \psi \in T$;
3. if $\neg(\varphi \wedge \psi) \in T$ then either $\neg\varphi \in T$ or $\neg\psi \in T$;
4. there is no formula ψ such that ψ and $\neg\psi$ are in T.

A set of formulas T is *blatantly inconsistent* if for some formula ψ, both ψ and $\neg\psi$ are in T.

In a tableau for a modal logic, for a given formula φ, $\mathrm{Sub}(\varphi)$ denotes the set of all subformulas of φ and $\neg\mathrm{Sub}(\varphi) = \mathrm{Sub}(\varphi) \cup \{\neg\psi : \psi \in \mathrm{Sub}(\varphi)\}$.

Definition 3.2. (TEAMLOG$^{\text{ind}}$ tableau)
A TEAMLOG^{ind} *tableau* $\mathcal{T}$ is a tuple

$$\mathcal{T} = (W, \{B_i \,:\, i \in \mathcal{A}\}, \{G_i \,:\, i \in \mathcal{A}\}, \{I_i \,:\, i \in \mathcal{A}\}, L),$$

where W is a set of states, B_i, G_i, I_i are binary relations on W, and L is a *labeling function* associating with each state $w \in W$ a set $L(w)$ of formulas, such that $L(w)$ is a propositional tableau. Here follow the two conditions that every modal tableau for our language must satisfy (see [16]):

1. If $\mathrm{BEL}(i, \varphi) \in L(w)$ and $(w, v) \in B_i$, then $\varphi \in L(v)$; similarly for $\mathrm{GOAL}(i, \varphi)$ w.r.t. G_i and $\mathrm{INT}(i, \varphi)$ w.r.t. I_i.
2. If $\neg\mathrm{BEL}(i, \varphi) \in L(w)$, then there exists a v with $(w, v) \in B_i$ and $\neg\varphi \in L(v)$; similarly for $\mathrm{GOAL}(i, \varphi)$ w.r.t. G_i and $\mathrm{INT}(i, \varphi)$ w.r.t. I_i.

Furthermore, a TEAMLOG$^{\text{ind}}$ tableau must satisfy the following additional conditions related to axioms of TEAMLOG$^{\text{ind}}$:

TA6	if $\mathrm{BEL}(i, \varphi) \in L(w)$, then either $\varphi \in L(w)$ or there exists $v \in W$ such that $(w, v) \in B_i$,
TA45	if $(w, v) \in B_i$ then $\mathrm{BEL}(i, \varphi) \in L(w)$ iff $\mathrm{BEL}(i, \varphi) \in L(v)$,
TA78$_{GB}$	if $(w, v) \in B_i$ then $\mathrm{GOAL}(i, \varphi) \in L(w)$ iff $\mathrm{GOAL}(i, \varphi) \in L(v)$,
TA6$_I$	if $\mathrm{INT}(i, \varphi) \in L(w)$, then either $\varphi \in L(w)$ or there exists $v \in W$ such that $(w, v) \in I_i$,
TA78$_{IB}$	if $(w, v) \in B_i$ then $\mathrm{INT}(i, \varphi) \in L(w)$ iff $\mathrm{INT}(i, \varphi) \in L(v)$,
TA9$_{IG}$	if $(w, v) \in G_i$ and $\mathrm{INT}(i, \varphi) \in L(w)$ then $\varphi \in L(v)$,

Condition **TA6** corresponds to belief consistency[2], **TA45** to positive and negative introspection of beliefs[3], **TA78**$_{GB}$ to positive and negative introspection of goals, **TA78**$_{IB}$ to positive and negative introspection of intentions, and **TA9**$_{IG}$ to goal–intention compatibility.

Given a formula φ we say that $\mathcal{T} = (W, \{B_i : i \in \mathcal{A}\}, \{G_i : i \in \mathcal{A}\}, \{I_i : i \in \mathcal{A}\}, L)$ is a TEAMLOG$^{\text{ind}}$ tableau for φ if $\mathcal{T}$ is a a TEAMLOG$^{\text{ind}}$ tableau and there is a state $w \in W$ such that $\varphi \in L(w)$.

Throughout further discussion we will use the notion of modal depth, which we define below (the definition is for the broader language of TEAMLOG).

Definition 3.3. (Modal depth)
Let φ be a TEAMLOG formula, then *modal depth* of φ, denoted by $\mathrm{dep}(\varphi)$ is defined inductively as follows:

- $\mathrm{dep}(p) = 0$, where $p \in \mathcal{P}$,
- $\mathrm{dep}(\neg\psi) = \mathrm{dep}(\psi)$,
- $\mathrm{dep}(\psi_1 \text{ op } \psi_2) = \max\{\mathrm{dep}(\psi_1), \mathrm{dep}(\psi_2)\}$, where $\text{op} \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$,
- $\mathrm{dep}(\mathrm{OP}(i, \psi)) = \mathrm{dep}(\psi) + 1$, where $\mathrm{OP} \in \{\mathrm{BEL}, \mathrm{GOAL}, \mathrm{INT}\}$,
- $\mathrm{dep}(\mathrm{OP}_G(\psi)) = \mathrm{dep}(\psi) + 1$, where $\mathrm{OP} \in \{\text{C-BEL}, \text{M-INT}\}$.

Let F be a set of TEAMLOG formulas. Then $\mathrm{dep}(F) = \max\{\mathrm{dep}(\psi) : \psi \in F\}$, if $F \neq \varnothing$, and $\mathrm{dep}(\varnothing) = 0$.

The following analogue to the proposition shown in [16] for $S5_n$ can be shown, giving a basis for an algorithm checking TEAMLOG$^{\text{ind}}$ satisfiability:

[2] This is a condition that occurs in [16], corresponding to the consistency axiom.

[3] We give this condition instead of two other conditions given in [16] as correspondents to positive and negative introspection axioms in a $KD45_n$ tableau. The given condition is exactly the condition the authors of [16] give, together with a condition corresponding to the truth axiom for $S5_n$.

Proposition 3.1. A formula φ is TEAMLOG$^{\text{ind}}$ satisfiable iff there is a TEAMLOG$^{\text{ind}}$ tableau for φ.

Proof:
The proof is very similar to the proof given in [16] for $S5_n$ tableaus. Although we have to deal with new conditions here, this is not a problem due to the similarity of conditions **TA78**$_{GB}$, **TA78**$_{IB}$ to condition **TA45**. The direction from left to right is a straightforward adaptation of the proof in [16], and we leave it to the reader.

When constructing a model for φ out of a tableau for φ in the right to left part we have to construct a "serial closure" of some relations. This is done by making isolated states accessible from themselves. For example accessibility relations I'_i for intentions would be defined on the basis of relations I_i in a tableau as follows: $I'_i = I''_i \cup \{(w, w) : \forall v \in W.\ (w, v) \notin I''_i\}$, where I''_i is the smallest set containing I_i and satisfying properties corresponding to axioms **A7**$_{IB}$ and **A8**$_{IB}$. □

3.1. The Algorithm for Satisfiability of TEAMLOG$^{\text{ind}}$

The algorithm presented below tries to construct, for a given formula φ, a *pre-tableau* – a tree-like structure that forms the basis for a TEAMLOG$^{\text{ind}}$ tableau for φ. Nodes of this pre-tableau are labelled with subsets of $\neg\mathrm{Sub}(\varphi)$. Nodes that are fully expanded propositional tableaus are called *states* and all other nodes are called *internal nodes*.

The algorithm is an adaptation of the algorithm presented in [16]. Modifications are connected with new axioms of the TEAMLOG$^{\text{ind}}$ logic and corresponding properties of accessibility relations.

Input: A formula φ.

Step 1 Construct a tree consisting of single node w, with $L(w) = \{\varphi\}$.

Step 2 Repeat until none of the steps **2.1** – **2.3** applies:

Step 2.1 Select a leaf s of the tree such that $L(s)$ is not blatantly inconsistent and is not a propositional tableau and select a formula ψ that violates the conditions of propositional tableau.

Step 2.1.1 If ψ is of the form $\neg\neg\xi$ then create a successor t of s and set $L(t) = L(s) \cup \{\xi\}$;

Step 2.1.2 If ψ is of the form $\xi_1 \wedge \xi_2$ then create a successor t of s and set $L(t) = L(s) \cup \{\xi_1, \xi_2\}$;

Step 2.1.3 If ψ is of the form $\xi_1 \vee \xi_2$ then create two successors t_1 and t_2 of s and set $L(t_1) = L(s) \cup \{\xi_1\}$ and $L(t_2) = L(s) \cup \{\xi_2\}$.

Step 2.2 Select a leaf s of the tree such that $L(s)$ is not blatantly inconsistent and is not a fully expanded propositional tableau and select $\psi \in L(s)$ with $\xi \in \mathrm{Sub}(\psi)$ such that $\{\xi, \neg\xi\} \cap L(s) = \varnothing$. Create two successors t_1 and t_2 of s and set $L(t_1) = L(s) \cup \{\xi\}$ and $L(t_2) = L(s) \cup \{\neg\xi\}$.

Step 2.3 Create successors of all states that are not blatantly inconsistent according to the following rules. Here, *s denotes a considered state* and the created successors will be called $\mathrm{b_i}$-, $\mathrm{g_i}$-, and $\mathrm{i_i}$-successors.

bel1 If $\mathrm{BEL}(i, \psi) \in L(s)$ and there are no formulas of the form $\neg\mathrm{BEL}(i, \chi) \in L(s)$, then let $L^{\mathrm{BEL}_i}(s) = \{\chi : \mathrm{BEL}(i, \chi) \in L(s)\} \cup \{\ \mathrm{OP}(i, \chi) : \mathrm{OP}(i, \chi) \in L(s)\}$, where $\mathrm{OP} \in$

$\{\mathrm{BEL}, \neg\mathrm{BEL}, \mathrm{GOAL}, \neg\mathrm{GOAL}, \mathrm{INT}, \neg\mathrm{INT}\}$. If there is no $\mathrm{b_i}$-ancestor t of s, such that $L^{\mathrm{BEL}_i}(t) = L^{\mathrm{BEL}_i}(s)$, then create a successor u of s (called $\mathrm{b_i}$-successor) with $L(u) = L^{\mathrm{BEL}_i}(s)$.

bel2 If $\neg\mathrm{BEL}(i, \psi) \in L(s)$, then let $L^{\neg\mathrm{BEL}_i}(s, \psi) = \{\neg\psi\} \cup L^{\mathrm{BEL}_i}(s)$. If there is no $\mathrm{b_i}$-ancestor t of s, such that $L^{\neg\mathrm{BEL}_i}(t, \psi) = L^{\neg\mathrm{BEL}_i}(s, \psi)$, then create a successor u of s (called $\mathrm{b_i}$-successor) with $L(u) = L^{\neg\mathrm{BEL}_i}(s, \psi)$.

int1 If $\mathrm{INT}(i, \psi) \in L(s)$ and there are no formulas of the form $\neg\mathrm{INT}(i, \chi) \in L(s)$, then let $L^{\mathrm{INT}_i}(s) = \{\chi : \mathrm{INT}(i, \chi) \in L(s)\}$. If there is no $\mathrm{i_i}$-ancestor t of s, such that $L^{\mathrm{INT}_i}(t) = L^{\mathrm{INT}_i}(s)$, then create a successor u of s (called $\mathrm{i_i}$-successor) with $L(u) = L^{\mathrm{BEL}_i}(s)$.

int2 If $\neg\mathrm{INT}(i, \psi) \in L(s)$, then let $L^{\neg\mathrm{INT}_i}(s, \psi) = \{\neg\psi\} \cup L^{\mathrm{INT}_i}(s)$. If there is no $\mathrm{i_i}$-ancestor t of s, such that $L^{\neg\mathrm{INT}_i}(t, \psi) = L^{\neg\mathrm{INT}_i}(s, \psi)$, then create a successor u of s (called $\mathrm{i_i}$-successor) with $L(u) = L^{\neg\mathrm{INT}_i}(s, \psi)$.

goal If $\neg\mathrm{GOAL}(i, \psi) \in L(s)$, then let $L^{\mathrm{GOAL}_i}(s) = \{\chi : \mathrm{GOAL}(i, \chi) \in L(s)\}$ and $L^{\neg\mathrm{GOAL}_i}(s, \psi) = \{\neg\psi\} \cup L^{\mathrm{GOAL}_i}(s) \cup L^{\mathrm{INT}_i}(s)$. If there is no $\mathrm{g_i}$-ancestor t of s, such that $L^{\neg\mathrm{GOAL}_i}(t, \psi) = L^{\neg\mathrm{GOAL}_i}(s, \psi)$, then create a successor u of s (called $\mathrm{g_i}$-successor) with $L(u) = L^{\neg\mathrm{GOAL}_i}(s, \psi)$.

Step 2.4 Mark a hitherto unmarked node 'satisfiable' if either it is a not blatantly inconsistent state and step **2.3** can not be applied to it and all its successors are marked 'satisfiable', or it is an internal node having at least one descendant marked 'satisfiable'.

Step 3 If the root is marked 'satisfiable' return 'satisfiable', otherwise return 'unsatisfiable'.

Before showing validity of the above algorithm, we will prove the following lemma which will be useful in further proofs. In what follows relations of $\mathrm{B_i}$-successor, $\mathrm{G_i}$-successor and $\mathrm{I_i}$-successor between states will be used and are defined as follows. Let s and t be a subsequent states. If t is a $\mathrm{b_i}$-, $\mathrm{g_i}$- or $\mathrm{i_i}$-successor of some node, then it is a $\mathrm{B_i}$-, $\mathrm{G_i}$- or $\mathrm{I_i}$-successor (respectively) of t.

Lemma 3.1. Let s and t be states of a pre-tableau constructed by the algorithm, such that t is a $\mathrm{B_i}$-successor s and t is not blatantly inconsistent. Then the following hold for $\mathrm{OP} \in \{\mathrm{BEL}, \mathrm{GOAL}, \mathrm{INT}\}$.

1. $L^{\mathrm{OP}_i}(s) = L^{\mathrm{OP}_i}(t)$.
2. $\neg\mathrm{OP}(i, \xi) \in L(s)$ and $L^{\neg\mathrm{OP}_i}(s, \xi) = L^{\neg\mathrm{OP}_i}(t, \xi)$, for any $\neg\mathrm{OP}(i, \xi) \in L(t)$.

Proof:
Note that if s has a $\mathrm{B_i}$-successor, then it is not blatantly inconsistent.

For point 1, let $\psi \in L^{\mathrm{OP}_i}(s)$. Then it is either $\mathrm{OP}(i, \psi) \in L(s)$ (and consequently $\mathrm{OP}(i, \psi) \in L(t)$) or $\mathrm{OP} = \mathrm{BEL}$, ψ is of the form $\mathrm{BEL}(i, \xi)$ and $\psi \in L(s)$ (consequently $\psi \in L(t)$). Thus $\psi \in L^{\mathrm{OP}_i}(t)$.

On the other hand, let $\psi \in L^{\mathrm{OP}_i}(t)$. Then either $\mathrm{OP}(i, \psi) \in L(t)$ or $\mathrm{OP} = \mathrm{BEL}$, ψ is of the form $\mathrm{BEL}(i, \xi)$ and $\psi \in L(t)$. Suppose that the first case holds. Since $L(s)$ is a fully expanded propositional tableau, either $\mathrm{OP}(i, \psi) \in L(s)$ or $\neg\mathrm{OP}(i, \psi) \in L(s)$. Because the second possibility leads to blatant inconsistency of $L(t)$ (as by the algorithm it implies that $\neg\mathrm{OP}(i, \psi) \in L(t)$), it must be that the first possibility holds and thus $\psi \in L^{\mathrm{BEL}_i}(s)$. The second case can be shown by similar arguments, as either $\mathrm{BEL}(i, \xi) \in L(s)$ or $\mathrm{BEL}(i, \xi) \in L(s)$.

For point 2, let $\neg\mathrm{OP}(i,\xi) \in L(t)$. Then by the fact that $L(s)$ is a fully expanded propositional tableau it is either $\neg\mathrm{OP}(i,\xi) \in L(s)$ or $\mathrm{OP}(i,\xi) \in L(s)$. As the second case leads to blatant inconsistency of $L(t)$, it must be the first one that holds.

$L^{\neg\mathrm{BEL}_i}(s,\xi) = L^{\neg\mathrm{BEL}_i}(t,\xi)$ can be shown by similar arguments to those used to show point 1 (note that $L^{\neg\mathrm{OP}_i}(v,\xi) = \{\neg\xi\} \cup L^{\mathrm{OP}_i}(v)$ or (in case of $\mathrm{OP} = \mathrm{GOAL}$) $L^{\neg\mathrm{OP}_i}(v,\xi) = \{\neg\xi\} \cup L^{\mathrm{OP}_i}(v) \cup L^{\mathrm{INT}_i}(v)$). □

Now we are ready to prove validity of the algorithm.

Lemma 3.2. For any formula φ the algorithm terminates.

Proof:
Let $|\varphi| = m$. For any node in a pre-tableau constructed by the algorithm $\mid L(s) \mid \leq 2m$ (if $L(s)$ is not blatantly inconsistent then $\mid L(s) \mid \leq m$). Any sequence of executions of steps **2.1** and **2.2** can have length $\leq m$. Thus on the path connecting any subsequent states s and t, there can be at most $m-1$ internal nodes.

If s and t are states such that t is a $\mathrm{G_i}$-successor or $\mathrm{I_i}$-successor of s then $\mathrm{dep}(L(t)) < \mathrm{dep}(L(s))$.

If t is a $\mathrm{B_i}$-successor of s and u is a $\mathrm{B_j}$-successor of t, where $i \neq j$, then $\mathrm{dep}(L(u)) < \mathrm{dep}(L(s))$.

If t is a $\mathrm{B_i}$-successor of s then, by lemma 3.1, t cannot have any $\mathrm{B_i}$-, $\mathrm{G_i}$- nor $\mathrm{I_i}$-successors. Thus, for any successor node u of t, $\mathrm{dep}(L(s)) < \mathrm{dep}(L(u))$.

All above arguments show that a pre-tableau constructed by the algorithm can have a depth at most $2 \cdot \mathrm{dep}(\varphi)m$. Since $\mathrm{dep}(\varphi) \leq m-1$, the modal depth of a pre-tableau is bounded by $m(m-1)$. This also shows that the algorithm terminates.

□

Lemma 3.3. A formula φ is satisfiable iff the algorithm returns 'satisfiable' on input φ.

Proof:
For the right to left direction, a tableau $\mathcal{T} = (W, \{B_i : i \in \mathcal{A}\}, \{G_i : i \in \mathcal{A}\}, \{I_i : i \in \mathcal{A}\}, L)$ based on the pre-tableau is constructed by the algorithm. W is the set of states of the pre-tableau. For $\{w,v\} \subseteq W$, let $(w,v) \in B_i^*$ if v is the closest descendant state of w and the first successor of w on the path between w and v is a $\mathrm{b_i}$–successor of w. Then B_i is determined as the transitive euclidean closure of the above relation B_i^*.

Relations G_i and I_i are defined analogically, but without taking the transitive Euclidean closure. Labels of states in W are the same as in the pre-tableau. Checking that $\mathcal{T}$ is a TEAMLOG$^{\mathrm{ind}}$ tableau is very much like in the case of $S5_n$ tableaus, with the new conditions **TA6**, **TA45**, **TA78**$_{GB}$, **TA6**$_I$, **TA78**$_{IB}$ being the most difficult cases.

For **TA6** note that if $v \in W$ has no successor states and $\mathrm{BEL}(i,\psi) \in L(v)$, then v cannot be a root, otherwise there is no ancestor of v such that its label is $L^{\mathrm{BEL}_i}(v)$, so step **2.3.bel1** of the algorithm applies to v and it cannot be a leaf. Therefore, there is $w \in W$, such that $(w,v) \in B_i$. Since $\mathrm{BEL}(i,\psi)$ is a subformula of φ, then either $\neg\mathrm{BEL}(i,\psi) \in L(w)$ or $\mathrm{BEL}(i,\psi) \in L(w)$. Because the first possibility leads to contradiction with $\mathrm{BEL}(i,\psi) \in L(v)$, then it must be the second, and this implies $\psi \in L(v)$.

Condition **TA**$_I$ can be shown similarly.

Condition **TA45** is also based on the fact that labels of states are fully expanded propositional tableaus, and can be shown similarly to **TA6** (see [16]). Since **TA78**$_{GB}$ and **TA78**$_{IB}$ are very similar to **TA45**, a b$_i$–successor inherits all formulas of the form $\mathrm{GOAL}(i, \psi)$, $\neg\mathrm{GOAL}(i, \psi)$, $\mathrm{INT}(i, \psi)$ and $\neg\mathrm{INT}(i, \psi)$, then they can be shown analogically to **TA45**. **Proposition 1** gives the final result.

For the left to right direction we show, for any node w in the pre-tableau, the claim that if w is not marked 'satisfiable' then $L(w)$ is inconsistent. From this it follows that if the root is not marked 'satisfiable' then $\neg\varphi$ is provable and thus φ is unsatisfiable.

The claim is shown by induction on the length of the longest path from a node w to a leaf of the pre-tableau. Most cases are easy and can be shown similarly to the case of $S5_n$ presented in [16]. We show only the most difficult case connected with new axioms of TEAMLOG$^{\mathrm{ind}}$, namely the one, in which w is not a leaf and has a b$_\mathrm{i}$–successor v generated by a formula of the form $\mathrm{BEL}(i, \psi) \in L(w)$ (other cases are either similar or easier). Since by induction hypothesis $L(v)$ is inconsistent, we can show using **A2**, **R1** and **R2** that the set $X = \{\mathrm{BEL}(i, \psi) : \mathrm{BEL}(i, \psi) \in L(w)\}) \cup \{\mathrm{BEL}(i, \psi) : \psi \in L(w)$ and is of the form $\mathrm{OP}(i, \chi)\}$ proves $\mathrm{BEL}(i, \bot)$, so by **A6** X is also inconsistent. Assume that $L(w)$ is consistent, then the set $Y = L(w) \cup \{\mathrm{BEL}(i, \psi) : \psi \in L(w)$ and is of the form $\mathrm{OP}(i, \chi)\} \cup \{\neg\mathrm{BEL}(i, \bot)\}$ is also consistent (by axioms **A4-6**, **A7-8**$_G B$ and **A7-8**$_I B$). This leads to contradiction, since $X \subseteq Y$, and thus $L(w)$ must be inconsistent. □

Theorem 3.1. The satisfiability problem for TEAMLOG$^{\mathrm{ind}}$ is PSPACE-complete

Proof:
Since the depth of the pre-tableau constructed by the algorithm for a given φ is at most $|\varphi|(|\varphi| - 1)$ and the algorithm is deterministic, it can be run on a deterministic Turing machine by depth-first search using polynomial space. Thus TEAMLOG$^{\mathrm{ind}}$ is in PSPACE. On the other hand the problem of KD_n satisfiability, known to be PSPACE-hard, can be reduced to TEAMLOG$^{\mathrm{ind}}$ satisfiability, so TEAMLOG$^{\mathrm{ind}}$ is PSPACE-complete. □

3.2. Effect of Bounding Modal Depth for TEAMLOG$^{\mathrm{ind}}$

As was shown in [15], bounding the modal depth of formulas by a constant results in reducing the complexity of the satisfiability problem for modal logics K_n, KD_n and $KD45_n$ to NP-complete.[4] An analogical result holds for the logic TEAMLOG$^{\mathrm{ind}}$, as we shall now show.

Theorem 3.2. For any fixed k, if the set of propositional atoms $\mathcal{P}$ is infinite and modal depth of formulas is bounded by k, then the satisfiability problem for TEAMLOG$^{\mathrm{ind}}$ is NP-complete.

Proof:
From the proof of **Lemma 3.2** we can observe that the number of states on a path from the root of a pre-tableau constructed by the algorithm to a leaf depends strictly on the modal depth of the input formula. Thus the size of the tableau corresponding to this pre-tableau is bounded by $\mathrm{O}(|\varphi|^{\mathrm{dep}(\varphi)})$. This means that the satisfiability of the formula φ with bounded modal depth can be checked by the following non-deterministic algorithm:

[4]Actually, in [15] logic T_n (not KD_n) is considered, but all proofs there that work for T_n work also for KD_n.

Input: A formula φ.

Step 1 Guess a tableau $\mathcal{T}$ satisfying φ.

Step 2 Check that $\mathcal{T}$ is indeed a tableau for φ.

Since the tableau $\mathcal{T}$ constructed at step **1** of the algorithm is of polynomial size, step **2** can be realized in polynomial time. Thus the satisfiability problem of TEAMLOG$^{\text{ind}}$ formula with modal depth bounded by a constant is in NP-time. It is also NP-complete, as the satisfiability problem for propositional logic is NP-hard. □

3.3. Effect of bounding the number of propositional atoms for TEAMLOG$^{\text{ind}}$

Another natural constraint on the language is bounding the number of propositional atoms. As was shown in [15], constraining the language of the logics K_n, KD_n (for $n \geq 1$) and $KD45_n$ (for $n \geq 2$) this way does not change the hardness of the satisfiability problem for them, even if $|\mathcal{P}| = 1$. This result holds also for our logic, as the formula used in proof of that fact in [15] could be expressed in TEAMLOG$^{\text{ind}}$ with use of the INT modality.

Similarly to [15] we can show that if bounding the number of propositional atoms is combined with bounding the modal depth of formulas, the complexity is reduced to linear time.

Theorem 3.3. For any fixed $k, l \geq 1$, if the number of propositional atoms is bounded by l and the modal depth of formulas is bounded by k, then the satisfiability problem for TEAMLOG$^{\text{ind}}$ can be solved in linear time.

Proof:
By the same argument as in [15], if $|\mathcal{P}| \leq l$, then there is only a finite number of equivalence classes (based on logical equivalence) of formulas of modal depth bounded by k in the language of TEAMLOG$^{\text{ind}}$. This can be proved by induction on k (see for example [3, Proposition 2.29]). Thus there is a finite set $\varphi_1, \ldots, \varphi_N$ of satisfiable formulas, each witness of a particular equivalence class all of whose members are satisfiable, and a corresponding fixed finite set of models $M_1, \ldots, M_N$ satisfying these formulas.

To check the satisfiability of a formula, it is enough to check whether it is satisfied in one of these models $M_1, \ldots, M_N$, and this can be done in time linear in the length of the formula; as the set of relevant models is fixed, it only contributes to the constant factor. □

4. Complexity of the System TEAMLOG

We will show that the satisfiability problem for the system TEAMLOG is EXPTIME-complete. First we prove that TEAMLOG has the small model property in the sense that for each satisfiable formula φ, a satisfying model of size $\mathcal{O}(2^{|\varphi|})$ can be found. To show this a filtration technique is used (see [3]). Let $G \subseteq \{1, \ldots, n\}$.

A set of formulas Σ closed for subformulas is *closed* if it satisfies the following:

Cl1 if $\text{C-BEL}_G(\varphi) \in \Sigma$, then $\text{E-BEL}_G(\varphi \wedge \text{C-BEL}_G(\varphi)) \in \Sigma$,

Cl2 if $\text{E-BEL}_G(\varphi) \in \Sigma$, then $\{\text{BEL}(j, \varphi) : j \in G\} \subseteq \Sigma$,

Cl3 if $\text{M-INT}_G(\varphi) \in \Sigma$, then $\text{E-INT}_G(\varphi \wedge \text{M-INT}_G(\varphi)) \in \Sigma$,

Cl4 if $\text{E-INT}_G(\varphi) \in \Sigma$, then $\{\text{INT}(j, \varphi) : j \in G\} \subseteq \Sigma$.

Let $\mathcal{M} = (W, \{B_i : i \in \mathcal{A}\}, \{G_i : i \in \mathcal{A}\}, \{I_i : i \in \mathcal{A}\}, \mathit{Val})$ be a TEAMLOG$^{\text{ind}}$ model, Σ a closed set, and let $\equiv_f^\Sigma \subseteq \Sigma \times \Sigma$ be an equivalence relation such that, for $\{w, v\} \subseteq W$, $w \equiv_f^\Sigma v$ iff for any $\varphi \in \Sigma$, $\mathcal{M}, w \models \varphi \Leftrightarrow \mathcal{M}, v \models \varphi$. Let $\mathcal{M}_\Sigma^f = (W^f, \{B_i^f : i \in \mathcal{A}\}, \{G_i^f : i \in \mathcal{A}\}, \{I_i^f : i \in \mathcal{A}\}, \mathit{Val}^f)$ be defined as follows:

F0 $W^f = W/\equiv_f^\Sigma$, $Val^f(p, [w]) = Val(p, w)$,

F1 $B_i^f = \{([w], [v])$: for any $\text{BEL}(i, \varphi) \in \Sigma,\ \mathcal{M}, w \models \text{BEL}(i, \varphi) \Rightarrow \mathcal{M}, v \models \varphi$ and for any $\text{OP}(i, \varphi) \in \Sigma,\ \mathcal{M}, w \models \text{OP}(i, \varphi) \Leftrightarrow \mathcal{M}, v \models \text{OP}(i, \varphi)\}$, where $\text{OP} \in \{\text{BEL}, \text{GOAL}, \text{INT}\}$.

F2 $G_i^f = \{([w], [v])$: for any $\text{GOAL}(i, \varphi) \in \Sigma,\ \mathcal{M}, w \models \text{GOAL}(i, \varphi) \Rightarrow \mathcal{M}, v \models \varphi$ and for any $\text{INT}(i, \varphi) \in \Sigma,\ \mathcal{M}, w \models \text{INT}(i, \varphi) \Rightarrow \mathcal{M}, v \models \varphi\}$.

F3 $I_i^f = \{([w], [v])$: for any $\text{INT}(i, \varphi) \in \Sigma,\ \mathcal{M}, w \models \text{INT}(i, \varphi) \Rightarrow \mathcal{M}, v \models \varphi\}$.

It is easy to check that if $\mathcal{M}$ is a TEAMLOG$^{\text{ind}}$ model, then so is $\mathcal{M}_\Sigma^f$ and, moreover, that if Σ is a closed set, then $\mathcal{M}_\Sigma^f$ is a *filtration* of $\mathcal{M}$ through Σ. This leads to the following standard lemma (thus left without a proof):

Lemma 4.1. If $\mathcal{M}$ is a TEAMLOG$^{\text{ind}}$ model and Σ is a closed set of formulas then for all $\varphi \in \Sigma$ and all $w \in W$, $\mathcal{M}, w \models \varphi$ iff $\mathcal{M}_\Sigma^f, [w] \models \varphi$.

From lemma 4.1 it follows that TEAMLOG has the finite model property and that its satisfiability problem is decidable. Let $\text{Cl}(\varphi)$ denote the smallest closed set containing $\text{Sub}(\varphi)$, and let $\neg\text{Cl}(\varphi)$ consist of all formulas in $\text{Cl}(\varphi)$ and their negations. If a formula φ is satisfiable then it is satisfiable in a filtration through $\text{Cl}(\varphi)$, and any such filtration has at most $|P(\text{Cl}(\varphi))| = \mathcal{O}(2^{|\varphi|})$ states.

Now we present an exponential time algorithm for checking TEAMLOG satisfiability of a formula φ. The algorithm and the proof of its validity are modified versions of the algorithm for checking satisfiability for PDL and its validity proof presented in [17][5]. The algorithm attempts to construct a model $\mathcal{M} = \mathcal{N}_{\text{Cl}(\varphi)}^f$, where $\mathcal{N}$ is a *canonical model* for TEAMLOG. This is done by constructing a sequence of models $\mathcal{M}^k$, being subsequent approximations of $\mathcal{M}$ as follows:

Input: A formula φ

Step 1 Construct a model $\mathcal{M}0 = (W0, \{B0_i : i \in \mathcal{A}\}, \{G0_i : i \in \mathcal{A}\}, \{I0_i : i \in \mathcal{A}\}, \mathit{Val}^0)$, where $W0$ is the set of all maximal subsets of $\neg\text{Cl}(\varphi)$, that is sets that for every $\psi \in \text{Cl}(\varphi)$ contain either ψ or $\neg\psi$, $\mathit{Val}^0(p, w) = 1$ iff $p \in w$, and accessibility relations are defined analogically as in $\mathcal{M}_{\text{Cl}(\varphi)}^f$. We present the definition of $B0_i$, which makes definitions **G1**, **I1** of $G0_i$ and $I0_i$ obvious:

[5]Note that one can see TEAMLOG as a modified and restricted version of PDL, where the BEL, GOAL and INT operators for each agent are seen as atomic programs satisfying some additional axioms, while group operators can be defined as complex programs using the $\cup$ and *.

B1 $B0_i = \{(w,v) :$ for any $\mathrm{BEL}(i,\varphi) \in \mathrm{Cl}(\varphi)$,
$\mathrm{BEL}(i,\varphi) \in w \Rightarrow \varphi \in v$ and for any $\mathrm{OP}(i,\varphi) \in \mathrm{Cl}(\varphi),\ \mathrm{OP}(i,\varphi) \in w \Leftrightarrow \mathrm{OP}(i,\varphi) \in v\}$,
where $\mathrm{OP} \in \{\mathrm{BEL}, \mathrm{GOAL}, \mathrm{INT}\}$.

Step 2 Construct a model $\mathcal{M}1$ by removing from $W0$ states that are not closed propositional tableaus.

Step 3 Repeat the following, starting with $k = 0$, until no state can be removed:

Step 3.1 Find a formula $\psi \in \neg\mathrm{Cl}(\varphi)$ and state $w \in W^k$ such that $\psi \in w$ and one of the conditions[6] below is not satisfied. It the state was found, remove it from W^k to obtain W^{k+1}.

AB1 if $\psi = \neg\mathrm{BEL}(i,\chi)$, then there exists $v \in B_i^k$ such that $\neg\chi \in v$ (analogical conditions **AG1** and **AI1** for GOAL and INT,

AB2 if $\psi = \mathrm{BEL}(i,\chi)$, then there exists $v \in B_i^k$ such that $\chi \in v$ (analogical condition **AI2** for INT),

AEB1 if $\psi = \neg\text{E-BEL}_G(i,\chi)$, then there exists $v \in B_G^k$ (where $B_G^k = \bigcup_{j\in G} B_j^k$ such that $\neg\chi \in v$ (analogical condition **AEI1** for E-INT_G),

ACB1 if $\psi = \neg\text{C-BEL}_G(i,\chi)$, then there exists $v \in (B_G^k)^*$ such that $\neg\chi \in v$ (analogical condition **AMI1** for M-INT_G).

Step 4 If there is a state in the model $\mathcal{M}^l$ obtained after step **3** containing φ, then return 'satisfiable', otherwise return 'unsatisfiable'.

It is obvious that the algorithm terminates. Moreover, since each step can be done in polynomial time, the algorithm terminates after $\mathcal{O}(2^{|\varphi|})$ steps. To prove the validity of the algorithm, we have to prove an analogue to a lemma in [17]. In the following lemma, $\mathrm{OP}_G \in \{\text{E-BEL}_G, \text{E-INT}_G\}$, $\mathrm{OP}_G^* \in \{\text{C-BEL}_G, \text{M-INT}_G\}$ and R denotes the relation corresponding to operator OP used in the particular context.

Lemma 4.2. Let $k \geq 1$ and assume that $\mathcal{M} \subseteq \mathcal{M}^{\,k}$. Let $\chi \in \mathrm{Cl}(\varphi)$ be such that every formula from $\mathrm{Cl}(\chi)$ of the form $\mathrm{OP}(i,\psi)$, $\mathrm{OP}_G(\psi)$ or $\mathrm{OP}_G^*(\psi)$ and $w \in W^k$ satisfies the conditions of step **3** of the algorithm. Then:

1 for all $\xi \in \mathrm{Cl}(\chi)$ and $v \in W^k$, $\xi \in v$ iff $\mathcal{M}, v \models \xi$;

2.1 for any $\mathrm{OP}(i,\xi) \in \mathrm{Cl}(\chi)$ and $\{w,v\} \subseteq W^k$:

2.1.a if $(w,v) \in R_i$ then $(w,v) \in R_i^k$;

2.1.b if $(w,v) \in R_i^k$ and $\mathrm{OP}(i,\xi) \in v$ then $\xi \in v$;

2.2 for any $\mathrm{OP}_G(\xi) \in \mathrm{Cl}(\chi)$ and $\{w,v\} \subseteq W^k$:

2.2.a if $(w,v) \in R_G$ then $(w,v) \in R_G^k$;

2.2.b if $(w,v) \in R_G^k$ and $\mathrm{OP}_G(\xi) \in v$ then $\xi \in v$;

[6]The conditions are analogical to conditions for PDL. The only differences are conditions **AB2** and **AI2** that correspond to axioms **A6** and **A6**$_I$.

2.3 for any $\mathrm{OP}^*_G(\xi) \in \mathrm{Cl}(\chi)$ and $\{w, v\} \subseteq W^k$:

2.3.a if $(w, v) \in (R_G)^*$ then $(w, v) \in (R^k_G)^*$;

2.3.b if $(w, v) \in (R^k_G)^*$ and $\mathrm{OP}^*_G(\xi) \in v$ then $\xi \in v$.

Proof:
The proof is analogical to the one of the lemma for PDL and the additional properties of TEAMLOG do not affect the argumentation. The proof of points **2.1-3** is essentially based on the fact that $\mathcal{M}$ is a filtration and similar techniques are used here to those from the proof of the filtration lemma. The proof of point **1** is by induction on the structure of ξ, similarly to its analogue for the lemma for PDL. □

Lemma 4.3. A formula φ is satisfiable iff the algorithm returns 'satisfiable' on input φ.

Proof:
Since every state $w \in W$ is a maximal subset of $\neg\mathrm{Cl}(\varphi)$, we have $W \subseteq W0$. Moreover, since every state $w \in W$ is a propositional tableau satisfying conditions from step **2** of the algorithm, thus $W \subseteq W1$. Conditions in step **2** also guarantee that no state $w \in W$ can be deleted in step **3**. This shows that $W \subseteq W^k$, for all W^k constructed throughout an execution of the algorithm. It follows that if φ is satisfiable then the algorithm will return 'satisfiable'.

If model $\mathcal{M}^l$ obtained after step **3** of the algorithm is not empty, then it can be easily checked that it is a TEAMLOG$^{\mathrm{ind}}$ model. This is because every model $\mathcal{M}^k$ constructed throughout an execution of the algorithm preserves conditions **B1, G1, I1**. Moreover, conditions **AB2**, **AI2** guarantee that relations B^l_i and I^l_i are serial. Now, if there is a $w \in W^l$ such that $\varphi \in w$, then (by **1** of lemma 4.2) $\mathcal{M}^l, w \models \varphi$. Since $\mathcal{M}^l$ is a TEAMLOG$^{\mathrm{ind}}$ model, then φ is TEAMLOG satisfiable. So the algorithm is valid. □

Theorem 4.1. The satisfiability problem for TEAMLOG is EXPTIME-complete.

Proof:
Immediately from lemmas 4.1, 4.2, and 4.3, it follows that satisfiability is in EXPTIME. It is also EXPTIME-hard by the same proof as used in theorem 4.2. □

Remark 4.1. The algorithm above and lemma 4.2 are kept similar to the ones presented in [17], so one can combine them to obtain a deterministic exponential time algorithm for a combination of TEAMLOG and PDL.

4.1. Effect of Bounding Modal Depth for TEAMLOG

The effect of bounding the modal depth of formulas on the complexity of the satisfiability problem for TEAMLOG is not as promising as in the case of TEAMLOG$^{\mathrm{ind}}$. It can be shown that even if modal depth is bounded by 2, the satisfiability problem remains EXPTIME-hard. The proof we give here is inspired by the proof of EXPTIME hardness of the satisfiability problem for PDL given in [3, Ch. 6.8].

Theorem 4.2. The satisfiability problem for deciding satisfiability of TEAMLOG formulas with modal depth bounded by 2 is EXPTIME-complete.

Proof:
The fact that satisfiablity for formulas of modal depth bounded by 2 is in EXPTIME follows immediately by theorem 4.1, because it is a special case. Thus, what we need to show is EXPTIME-hardness of the problem. To do this we will use the two-person corridor tiling game.

A tile is a 1×1 square, with fixed orientation and a color assigned to each side. There are two players taking part in the game and a referee who starts the game. The referee gives the players a finite set of $\{T_1, \ldots, T_s\}$ of tile types. Players will use tiles of these types to arrange them on the grid in such a way that the colors on the common sides of adjacent tiles match. Additionally there are two special tile types T_0 and T_{s+1}. T_0 is an all sides white type, used merely to mark the boundaries of the corridor inside which the two players will place their tiles. T_{s+1} is a special winning tile that can be placed only in the first column.

At the start of the game the referee fills in the first row (places $\{1, \ldots, m\}$) of the corridor with m initial tiles of types $\{T_1, \ldots, T_s\}$ and places two columns of T_0 type tiles in columns 0 and $m+1$ marking the boundaries of the corridor. Now the two players A and B place their tiles in alternating moves. Player A is the one to start. The corridor is to be filled row by row from bottom to top and from left to right. Thus the place of the next tile is determined and the only choice the players make is the type of tile to place. The color of a newly placed tile must fit the colors of its adjacent tiles. We will use $C(T', T, T'')$ to denote that T can be placed to the right of T' and above tile T'', thus that $right(T') = left(T)$ and $top(T'') = bottom(T)$, where $right$, $left$, top and $bottom$ give the colors of respective sides of a tile.

If after finitely many rounds a tiling is constructed in which a tile of type T_{s+1} is placed in the first column, then player A wins. Otherwise, that is if no player can make a legal move or if the game goes on infinitely long and no tile of type T_{s+1} is placed in the first column, player B wins. The problem of deciding if for a given setting of the game there is a winning strategy for player A is an EXPTIME-hard problem [6]. Following [3, Ch. 6.8] we will show that this problem can be reduced to the satisfiability problem of TEAMLOG formulas of modal depth ≤ 2.

In the proof of [3, Ch. 6.8] a formula is constructed for a given tiling game, such that a model of it is the game tree for given settings of the game with its root as a current state. States of the tree contain information about the actual configuration of the tiles, the player who is to move next, and the position at which the next tile is to be placed. The depth of the tree is bounded by m^{s+2}. Note that after m^{s+2} rounds, repetition of rows must have occurred and if A can win a game with repetitions, A can also win a game without them, thus it is enough to consider m^{s+2} rounds only.

The formula from the proof of [3, Ch. 6.8] uses two PDL modalities $[a]$ and $[a^*]$ and its depth is bounded by 2. These modalities could be replaced by $\text{M-INT}'_{\{1\}}$, where $\text{M-INT}'_G(\varphi)$ is a shortcut for $\text{M-INT}'_G(\varphi) \wedge \varphi$ (recall that $[a^*]$ is reflexive and M-INT is not), and $\text{INT}(1, \cdot)$. The proof would remain the same. Thus it can be shown that even if we consider M-INT with $n \geq 1$ and formulas with modal depth bounded by 2, the satisfiability problem remains in EXPTIME. Below we show a slightly modified version of the [3, Ch. 6.8] proof, adapted for C-BEL. In this case $n \geq 2$ is required. This is not surprising, as for $n = 1$ C-BEL is equivalent to BEL because by axioms **A4** and **A5**, $\text{BEL}(1, \varphi)$ and $\text{BEL}(1, \text{BEL}(1, \varphi))$ are equivalent.

Let $\mathcal{G} = (m, \mathcal{T}, (I_1, \ldots, I_m))$, where $\mathcal{T} = \{T_0, \ldots, T_{s+1}\}$ and $I_j \in \mathcal{T}$ for $0 \leq j \leq m$, be a setting for a two person corridor tiling game described above. Here, $(I_1, \ldots, I_m)$ is the row of types of the initial tiling of the first row of the corridor. We construct a formula $\varphi(\mathcal{G})$ such that it is satisfiable iff player A has a winning strategy. The following propositional symbols are used to construct a formula:

- a to indicate that A has the next move; we will also use p_1 to denote a and p_2 to denote $\neg a$ in order to shorten some formulas,
- $pos_1, \ldots, pos_m$ to indicate the column in which a tile is to be placed in the current round,
- $col_i(T)$, for $0 \leq i \leq m+1$ and $T \in \mathcal{T}$, to indicate that a tile previously placed in column i is of type T,[7]
- win to indicate that the current position is a winning position for A,
- $q_1, \ldots q_N$, where $N = \lceil \log_2 (m^{s+2}) \rceil$, to enumerate states; boolean values of these variables in a given state can be treated as a representation of a binary number with q_1 being the least significant bit and q_N being the most significant one; we will give the same number to all states belonging to the same round; we will use the notation $round = k$ as a shortcut for the formula expressing that the number encoded by $q_N \ldots q_1$ is equal to k.

The formula $\varphi(\mathcal{G})$ will be composed of the following formulas describing settings of the game and giving necessary and sufficient conditions for the existence of a winning strategy for A. In what follows, $k \in \{1, 2\}$, $0 \leq i \neq j \leq m+1$, $0 \leq x \neq y \leq s+1$ and $\{T, T', T''\} \subseteq \mathcal{T}$ (if not stated differently). We will also use C-BEL$'_G(\varphi)$ as a shortcut for C-BEL$_G(\varphi) \wedge \varphi$. We also use the conventions that $\bigwedge \varnothing = \top$ and $\bigvee \varnothing = \bot$.

$$a \wedge pos_1 \wedge col_0(T_0) \wedge col_{m+1}(T_0) \wedge col_1(I_1) \wedge \ldots \wedge col_m(I_m) \tag{1}$$

$$\text{C-BEL}_{\{1,2\}}(pos_1 \vee \ldots \vee pos_m) \tag{2}$$

$$\text{C-BEL}'_{\{1,2\}}(pos_i \rightarrow \neg pos_j),\ 1 \leq i \neq j \leq m \tag{3}$$

$$\text{C-BEL}_{\{1,2\}}(col_i(T_0) \vee \ldots \vee col_i(T_{s+1})) \tag{4}$$

$$\text{C-BEL}'_{\{1,2\}}(col_i(T_x) \rightarrow \neg col_i(T_y)) \tag{5}$$

$$\text{C-BEL}_{\{1,2\}}(col_0(T_0) \wedge col_{m+1}(T_0)) \tag{6}$$

$$\begin{aligned} \text{C-BEL}'_{\{1,2\}}((\neg pos_i \rightarrow ((col_i(T_x) \rightarrow \text{BEL}(k, col_i(T_x))) \wedge \\ (\neg col_i(T_x) \rightarrow \text{BEL}(k, \neg col_i(T_x))))), \end{aligned} \tag{7}$$

$$\begin{aligned} \text{C-BEL}'_{\{1,2\}}((pos_m \wedge p_k \rightarrow \text{BEL}(k, pos_1)) \wedge \\ (pos_1 \wedge p_k \rightarrow \text{BEL}(k, pos_2)) \wedge \ldots \wedge (pos_{m-1} \wedge p_k \rightarrow \text{BEL}(k, pos_m)))), \end{aligned} \tag{8}$$

[7]Note that $col_i(T)$ is a parametrized name of a propositional symbol.

$$\text{C-BEL}'_{\{1,2\}}((a \to \text{BEL}(1, \neg a)) \wedge (\neg a \to \text{BEL}(2, a))) \tag{9}$$

$$\begin{aligned}\text{C-BEL}'_{\{1,2\}}\Big(&pos_i \wedge col_{i-1}(T') \wedge col_i(T'') \wedge p_k \to \\ &\text{BEL}\left(k, \bigvee\{col_i(T) : C(T', T, T'')\}\right)\Big),\ 1 \le i \le m,\end{aligned} \tag{10}$$

$$\text{C-BEL}'_{\{1,2\}}\left(pos_n \to \text{BEL}\left(k, \bigvee\{col_n(T) : right(T) = white\}\right)\right) \tag{11}$$

$$\begin{aligned}&\text{C-BEL}'_{\{1,2\}}\Big(\neg a \wedge pos_i \wedge col_i(T'') \wedge col_{i-1}(t') \to \\ &\bigwedge\{\neg\text{BEL}(k, \neg col_i(T)) : C(T', T, T'')\}\Big),\ 1 \le i \le m\end{aligned} \tag{12}$$

$$\begin{aligned}win \wedge \text{C-BEL}'_{\{1,2\}}(win \to (col_1(T_{s+1}) \vee (a \wedge \neg\text{BEL}(1, \neg win))\vee \\ (\neg a \wedge \text{BEL}(2, win)))))\end{aligned} \tag{13}$$

$$\text{C-BEL}'_{\{1,2\}}((round = N) \to \text{BEL}(k, \neg win)) \tag{14}$$

Formulas (1–7) describe the settings of the game. The initial setting is as described by (1). During the game tiles are placed in exactly one of the columns $1..m$ (2–3) and in every column exactly one tile type was previously placed (4–5). The boundary tiles are placed in columns 0 and $m+1$ (6) and nothing changes in columns where no tile is placed during the game (7).

Formulas (8–11) describe the rules of the game. Tiles are placed from bottom to top, row by row from left to right (8); thus, the first conjunct of (8) represents the flipping of one row to the next. The players alternate (9). Tiles that are placed have to match adjacent tiles (10–11). The formula (12) ensures that all possible moves by player B are encoded in the model.

Formula (13) gives properties of states that can be marked as winning positions for the player A and formula (14) states that all states reached after $\geq N$ rounds can not be winning positions for A. Similarly to [3, Lemma 6.51], one can force exponentially deep models of TEAMLOG for satisfying some specific formulas of depth ≤ 2. Specifically, to enumerate the states according to rounds of the game we will need the following additional formula.

$$\bigwedge_{j=1}^{N} \neg q_j \wedge \text{C-BEL}'_{\{1,2\}}\left(INC_0 \wedge \bigwedge_{j=1}^{N-1} INC_1(j)\right), \tag{15}$$

where

$$\begin{aligned}INC_0 \equiv \neg q_1 \to \Big(&\text{BEL}(1, q_1)\wedge \\ &\bigwedge_{j=2}^{N}((q_j \to \text{BEL}(1, q_j)) \wedge (\neg q_j \to \text{BEL}(1, \neg q_j))\Big)\end{aligned} \tag{16}$$

$$
\begin{aligned}
INC_1(i) \equiv & \Big(\neg q_{i+1} \wedge \bigwedge_{j=1}^{i} q_j\Big) \rightarrow \mathrm{BEL}\Big(2, q_{i+1} \wedge \bigwedge_{j=1}^{i} \neg q_j \wedge \\
& \bigwedge_{j=i+2}^{N} ((q_j \rightarrow \mathrm{BEL}(2, q_j)) \wedge (\neg q_j \rightarrow \mathrm{BEL}(2, \neg q_j)))\Big)
\end{aligned} \tag{17}
$$

Formula (15) enforces that the root of the model receives a number $(0\ldots0)_2$ and worlds corresponding to states in subsequent rounds of the game receive subsequent numbers in binary representation. The formula INC_0 is responsible for increasing even numbers, and $INC_1(i)$ is responsible for increasing odd numbers ending with a sequence of i 1 and having 0 at the position $i+1$.

The formula $\varphi(\mathcal{G})$ is the conjunction of formulas (1–15) and it is of size polynomial with respect to m. It can be easily seen that if A has a winning strategy in the particular game, the formula $\varphi(\mathcal{G})$ is satisfiable in a model built on the basis of a game tree for this game. Edges corresponding to turns of player A are the basis for accessibility relation B_1 and those corresponding to turns of player B are the basis for accessibility relation B_2. To satisfy the properties of the model, B_1 and B_2 are extended by identity in worlds that violate the seriality property. All other relations B_i and I_i are set to identity and G_i are set to $\varnothing$. Valuation of propositional variables in the worlds of the model is automatically determined by the description of the situation in the corresponding states of the game.

On the other hand, if $\varphi(\mathcal{G})$ is satisfiable, A can use a model of $\varphi(\mathcal{G})$ as a guide for his winning strategy. At the beginning, he chooses a transition (represented by accessibility relation B_1) to a world where win is true, and plays accordingly. Player A does analogically in all subsequent rounds of the game. He can track the worlds corresponding to states of subsequent rounds of the game, by following relations B_1 and B_2 alternatingly. Notice for all worlds v corresponding to states where A is to play and where A has a winning strategy (that is win is true) it must be $(v, v) \notin B_1$, as guaranteed by (9). The same holds for B_2 and states where B is to play. Notice also that formula (14) guarantees that A will reach a winning position in a finite number of steps if he plays as described above. □

4.2. Effect of bounding the number of propositional atoms for TeamLog

If the number of propositional atoms is bounded by 1, the complexity of satisfiability problem for logic TeamLog remains EXPTIME-hard. It can be easily shown by using an analogical technique to that described in [15]. The idea is to substitute propositional symbols used in the proof of the theorem 4.2, by so called pp-like formulas, that would have similar properties as propositional atoms (in terms of independence of their valuations in the worlds of a model). Suppose that propositional atoms are denoted by q_j. Then a pp-like formula replacing the propositional symbol q_j is $\neg\mathrm{OP}(k, \neg p \wedge \neg\mathrm{BEL}^j(1, \neg p))$, where $\mathrm{OP}(k, \cdot)$ is any modal operator not used in the proof of theorem 4.2. [8] See [15] for additional details and an extended discussion of using pp-like formulas.

Similarly to the case of TeamLog$^{\mathrm{ind}}$ we can show that if bounding the number of propositional atoms is combined with bounding the modal depth of formulas, the complexity is reduced to linear time. The proof is analogical to this for TeamLog$^{\mathrm{ind}}$.

[8] Note that this argument will not work for the logic KD_1 with group operator M-INT, nor will it work for the logic $KD45_2$ with group operator C-BEL, because there is no "free" modal operator left to be used as $\mathrm{OP}(k, \cdot)$ for these cases. We do not know yet what would be the complexity of the satisfiability problem for these logics, when the number of propositional atoms is bounded.

Theorem 4.3. For any fixed $k, l \geq 1$, if the number of propositional atoms is bounded by l and the modal depth of formulas is bounded by k, then the satisfiability problem for TEAMLOG can be solved in linear time.

5. Discussion and Conclusions

This paper deals with the complexity of two important components of TEAMLOG. The first one covers agents' individual attitudes, including their interdependencies, which makes the decision procedure more complicated. The second one deals with the team attitude par excellence: collective intention. Importantly, however, our results have a more general impact. The tableau methods we use can be adapted to the non-temporal parts of other multi-modal logics which are similar in spirit to ours, such as the KARO framework [1].

The presented system defining collective intentions is decidable. More precisely, as proved in the current paper, it is EXPTIME-complete. As with other modal logics, an option would be to develop a variety of different algorithms and heuristics, each performing well on a limited class of inputs. For example, it is known that restricting the number of propositional atoms to be used and/or the depth of modal nesting may reduce the complexity (cf. [15, 18]). We explored these possibilities in this paper for both individual and collective part of TEAMLOG. Also, when considering specific applications, it is possible to reduce some of the infinitary character of collective beliefs and intentions to more manageable proportions (cf. [14, Ch. 11]). From the perspective of AI applications it is particularly interesting to restrict considerations to Horn-like formulas (see, [22]). Such restrictions are essential when the strongest motivational attitude, collective commitment, is considered [11] in order to produce system specifications of lower complexity.

Another promising technique to reduce the complexity could depend on simplifying multimodal theories of collective attitudes of agents using approximations in the spirit of rough set theory introduced by Pawlak [23, 24]. His influential ideas, developed over the last 25 years by many researchers, appeared very useful, among others, in the context of reducing the complexity of reasoning over large data sets. It seems rather natural to extend his techniques to the context of approximating the logical theories in question. In fact, logical approximations have been considered in papers [20, 5, 21, 8] and in a book [7]. It can be shown that the approximations considered in [21, 8] are as strong as rough approximations introduced by Pawlak.

6. Acknowledgments

Barbara Dunin-Kęplicz and Rineke Verbrugge are grateful to the Netherlands Institute for Advanced Study (NIAS) for providing an opportunity to complete this paper as Fellows-in-Residence. We would also like to thank the anonymous referees for helpful comments and suggestions.

References

[1] Aldewereld, H., van der Hoek, W., Meyer, J.-J. C.: Rational Teams: Logical Aspects of Multi-Agent Systems, *Fundamenta Informaticae*, **63 (2,3)**, 2004, 159–183.

[2] van Benthem, J.: Correspondence Theory, in: *Handbook of Philosophical Logic* (D. Gabbay, F. Guenthner, Eds.), vol. 2, Reidel, Dordrecht, 1984, 167–247.

[3] Blackburn, P., de Rijke, M., Venema, Y.: *Modal Logic*, vol. 53 of *Cambridge Tracts in Theoretical Computer Science*, Cambridge University Press, 2002, ISBN 0-521-52714-7.

[4] Blackburn, P., Spaan, E.: A modal perspective on the computational complexity of attribute value grammar, *Journal of Logic, Language and Information*, **2**, 1993, 129–169.

[5] Cadoli, M.: *Tractable Reasoning in Artificial Intelligence*, Number 941 in Lecture Notes in Artificial Intelligence, Springer, 1995.

[6] Chlebus, B.: Domino-tiling games, *Journal of Computer and System Sciences*, **32**, 1986, 374–392.

[7] Doherty, P., Łukaszewicz, W., Skowron, A., Szałas, A.: *Knowledge Representation Techniques. A Rough Set Approach*, vol. 202 of *Studies in Fuziness and Soft Computing*, Springer Verlag, 2006.

[8] Doherty, P., Łukaszewicz, W., Szałas, A.: Computing Strongest Necessary and Weakest Sufficient Conditions of First-Order Formulas, *International Joint Conference on AI (IJCAI'2001)*, 2000, 145 – 151.

[9] Dunin-Kęplicz, B., Verbrugge, R.: Collective Intentions, *Fundamenta Informaticae*, **51(3)**, 2002, 271–295.

[10] Dunin-Kęplicz, B., Verbrugge, R.: Evolution of Collective Commitments During Teamwork, *Fundamenta Informaticae*, **56**, 2003, 329–371.

[11] Dunin-Kęplicz, B., Verbrugge, R.: A Tuning Machine for Cooperative Problem Solving, *Fundamenta Informaticae*, **63**, 2004, 283–307.

[12] Dunin-Kęplicz, B., Verbrugge, R.: Creating Common Beliefs in Rescue Situations, *Monitoring, Security, and Rescue Techniques in Multiagent Systems* (B. Dunin-Kęplicz, A. Jankowski, A.Skowron, M.Szczuka, Eds.), Advances in Soft Computing, Springer Verlag, Berlin, 2005.

[13] Dziubiński, M., Verbrugge, R., Dunin-Kęplicz, B.: Complexity of a theory of collective attitudes in teamwork, *International Conference on Intelligent Agent Technology (IAT 2005)*, IEEE Computer Society, Los Alamitos (CA), 2005.

[14] Fagin, R., Halpern, J., Moses, Y., Vardi, M.: *Reasoning about Knowledge*, MIT Press, Cambridge, MA, 1995.

[15] Halpern, J. Y.: The effect of bounding the number of primitive propositions and the depth of nesting on the complexity of modal logic, *Artificial Intelligence*, **75**(3), 1995, 361–372, ISSN 0004-3702.

[16] Halpern, J. Y., Moses, Y.: A guide to completeness and complexity for modal logics of knowledge and belief, *Artificial Intelligence*, **54**(3), 1992, 319–379, ISSN 0004-3702.

[17] Harel, D., Kozen, D., Tiuryn, J.: *Dynamic Logic*, MIT Press, Cambridge, MA, 2000.

[18] Hustadt, U., Schmidt, R.: On Evaluating Decision Procedures for Modal Logics, *Proceedings IJCAI'97* (M. Pollack, Ed.), Morgan Kauffman, Los Angeles (CA), 1997.

[19] Kacprzak, M., Lomuscio, A., Penczek, W.: From Bounded to Unbounded Model Checking for Interpreted Systems, *Fundamenta Informaticae*, **63 (2,3)**, 2004, 107–308.

[20] Kautz, H., Selman, B.: Knowledge Compilation and Theory Approximation, *Journal of the ACM*, **43**(2), 1996, 193–224.

[21] Lin, F.: On strongest Necessary and Weakest sufficient Conditions, *Proc. 7th International Conf. on Principles of Knowledge Representation and Reasoning, KR2000* (A. Cohn, F. Giunchiglia, B. Selman, Eds.), Morgan Kaufmann Pub., Inc., 2000.

[22] Nguyen, L. A.: On the Complexity of Fragments of Modal Logics, *Advances in Modal Logic 5, Papers from the Fifth Conference on "Advances in Modal logic," held in Manchester (UK) in September 2004* (R. Schmidt, I. Pratt-Hartmann, M. Reynolds, H. Wansing, Eds.), King's College Publications, 2004.

[23] Pawlak, Z.: Information systems – theoretical foundations, *Information Systems*, **6**, 1981, 205–218.

[24] Pawlak, Z.: *Rough Sets. Theoretical Aspects of Reasoning about Data*, Kluwer Academic Publishers, Dordrecht, 1991.