Checking Agent Intentions in Games

Nathalie Chetcuti-Sperandio, Alix Goudyme, Tiago de Lima, Sylvain Lagrue

Checking Agent Intentions in Games

Nathalie Chetcuti-Sperandio
CRIL
Univ. Artois & CNRS
Lens, France
chetcuti@cril.fr

Alix Goudyme
CRIL
Univ. Artois & CNRS
Lens, France
goudyme@cril.fr

Sylvain Lagrue
Université de Technologie de Compiègne
CNRS, Heudiasyc
Compiègne, France
sylvain.lagrue@hds.utc.fr

Tiago de Lima
CRIL
Univ. Artois & CNRS
Lens, France
delima@cril.fr

***Abstract*—Rational agents' decisions are driven by their intentions, in the sense that agents execute actions that most probably lead to situations where their intentions are achieved. Using that insight, this paper proposes a method for 'intention checking': let a description of a game, a state and the action executed by the agent at that state be given, the method checks whether the agent acted with the intention to reach a situation where some proposition 'p' is true. We use a logic with epistemic and temporal operators to reason about games and extend it with an intention operator 'IX'. Formulas of the form 'IX(p)' are defined to be true in the situations where the intention check method verifies that the agent acts with the intention to achieve 'p' in the next state of the game. We show that this operator satisfies the principles of Bratman's Asymmetry Thesis, and we also compare it to other theories of intention.**

***Index Terms*—Knowledge Representation and Reasoning, Intention, Alternating-time Temporal Logic**

I. Introduction

The concept of intention has been widely studied in artificial intelligence. It is one of the main components of BDI agent architectures [1]–[3] and a vast literature focuses on its logical properties and its links with actions, beliefs and desires. Broadly-accepted theories of intention (e.g., [4], [5]) stipulate that, unlike desires, intentions are rational attitudes: agents do not intend to achieve what they know to be impossible. Games in general and strategic games in particular, offer a controlled environment where the goal of a rational agent is to win, or, at least, to maximize utility. We take into consideration in this paper both games with perfect and imperfect information. In such games, rational agents' decisions are driven by their intentions in the sense that agents execute actions that most probably lead to situations where their intentions are achieved. Therefore, it is conceivable that, by observing agents' behaviour, one could use logical reasoning to calculate the intentions of rational agents.

Different attempts have been made to link intentions, strategies and games, for instance [6]–[8], or to define intentional agents for a specific game [9]. None of them can be used to analyze the intention of a rational agent *post hoc*. Indeed, in such approaches, agent intentions are directly provided in the model. In our approach, agent intentions are supposed to be unknown and have to be determined.

This work has been partially supported by the AI Chair project Responsible AI (ANR-19-CHIA-0008) from the French National Agency of Research.

The aim of this paper is to propose logical foundations for intention checking: let be given a description of a game, a state and the action executed by the agent at that state, our method checks whether the agent acted with the intention to reach a situation where some proposition p is true. Intentions are to be calculated given the information one *can* obtain from the game, namely, what agents know in the game, game objectives and how the agents played the game. We can determine agents' knowledge in the game by analyzing their initial knowledge and the information they obtain during the game.

In the present contribution, we first propose a definition for the concept of intention in games. Then, we use a logic with epistemic and temporal operators to reason about games and extend it with an intention operator '$\mathrm{I}_i^a\mathrm{X}^k$'. Formulas of the form $\mathrm{I}_i^a\mathrm{X}^k\phi$ are defined to be true in the exact situations where the aforementioned method verifies that, when i executes a, she intends to achieve ϕ after k steps. And finally, we compare this operator with other logical theories of intention, such as Shoham's database perspective [5], and we also show that it satisfies all principles of Bratman's Asymmetry Thesis [4], as well as Cohen and Levesque's side-effect principle [10].

Our proposition can be used to provide a tool that helps an expert understand the behaviour of a powerful game program that cannot give an explanation of its choices (for example, the moves played by AlphaZero [11]). The expert could, by having access to the trace of the game, perform queries such as 'did the program play this move because it wished that particular property to be true k turns later' and thus exhibit the tactical (even strategic) reasoning of the program.

The remainder of this article is organized as follows. The next section (Sec. II) introduces the basic logical formalism used throughout the paper. Section III defines the intention checking method and explains it using various examples. Section IV proposes an extension of the basic formalism and studies a number of properties satisfied (or not) by our theory of agent intention. The last section (Sec. V) concludes and points out some perspectives.

II. Preliminaries

A standard (game theoretic) definition of extensive-form game can be found in several textbooks (e.g., [12], [13]). However, logical theories of intention are usually formalized in some kind of dynamic logic [10], temporal logic [1], [14], or situation calculus [15]. The underlying logic must suit the final purpose, but the particular syntax and semantics used is not the main contribution we intend to make here.

We chose to use imperfect information concurrent game structures (ICGS) [16] as models. These are an extension of the concurrent game structures (CGS) proposed by [17] to give semantics to alternating-time temporal logic (ATL). They extend CGS with a family of equivalence relations $\sim_i$ between states to model the information available to each agent i. If two states are related by $\sim_i$, then agent i cannot distinguish them.

An ICGS is close to an imperfect information extensive-form game, but there are three important differences. The first one is that ICGS permit concurrent actions, meaning that all agents act in every state of the game. Nonetheless, turn-based games can be represented: if it is not agent i's turn to play, then it is possible to specify that the only executable action for i at the given state is a "noop" (no-operation) action that does not change anything. The second difference is that ICGS do not have terminal states, i.e., all states have at least one executable action for each agent. To fill this gap, we allow terminal states in our models. The third difference is that ICGS do not have utilities. Again, we fill the gap by adding utility functions to our models.

Definition 1 (Game signature): A game signature is a triplet (P, N, A), where:

- P is a nonempty countable set of propositional variables;
- N is a nonempty finite set of agents (or players);
- A is a nonempty finite set of atomic actions (sometimes called moves or choices) available to agents.

Definition 2 (Model): Let a signature (P, N, A) be given. A model is a tuple $\mathcal{M} = (W, \pi, d, \delta, \sim, u)$, where:

- W is a nonempty finite set of states;
- $\pi : W \to 2^P$ is a valuation function;
- $d : N \times W \to 2^A$ is function assigning a set of legal actions to each agent at each state of the model;
- $\delta : W \times A^N \nrightarrow W$ is a (partially defined) transition function;
- $\sim : N \to 2^{W \times W}$ is a family of equivalence relations specifying the indistinguishable states to each agent;
- $u : N \times W \times A \nrightarrow \mathbb{R}$ is a (partially defined) utility function assigning a real value to the execution of each legal action by each agent at each state of the model.

A pointed model is a pair $(\mathcal{M}, w)$, where $\mathcal{M}$ is a model and w is a state in $\mathcal{M}$.

Note that our utility function is different from the one in standard extensive-form games, where it is defined for terminal states. Here, utilities are supposed to simulate the output of an evaluation function of a game program. Examples of well-known evaluation functions are minimax, alpha-beta, and monte-carlo tree search algorithms. Moreover, some game programs use heuristic functions to evaluate game positions. Since that is the way such programs reason about the game, when checking their intentions, u should implement the corresponding heuristic function.

To increase readability, we use several notational shortcuts in the remainder of the paper: $d_i(w)$ denotes $d(i, w)$; $D(w)$ denotes the set of legal joint actions at w, i.e., $D(w) = \{\alpha \in A^N \mid \alpha(i) \in d_i(w), \text{ for all } i \in N\}$; $\delta_i(w, a)$ denotes the set of states that the execution of action a by agent i can lead to, i.e., $\delta_i(w, a) = \{\delta(w, \alpha) \in W \mid \alpha \in D(w) \text{ and } \alpha(i) = a\}$; $w \sim_i w'$ denotes $(w, w') \in \sim(i)$; $[w]_i$ denotes the set of states that agent i cannot distinguish from w, i.e., $[w]_i = \{w' \mid w \sim_i w'\}$; and finally, $u_i(w, a)$ denotes $u(i, w, a)$.

We assume that models also satisfy the following constraints (both also present in [16]):

$$\delta(w, \alpha) \text{ is defined if and only if } \alpha \in D(w) \quad \text{(EXE)}$$

$$\text{If } w \sim_i w' \text{ then } d_i(w) = d_i(w') \quad \text{(CKA)}$$

The executability constraint (EXE) stipulates that, at each state, the legal joint actions are those such that each individual action is legal for each agent. This implies that agents choices are independent from each other. In other words, the action chosen by one agent cannot restrict the actions available to another agent.

The complete knowledge about (available) actions constraint (CKA) (name given in [18]) means that the executable actions available to each agent are the same among indistinguishable states. This must be so, otherwise agents could distinguish such states by the decisions they can make. It may seem very restrictive. However, we do not know of a "real" game where this constraint is not satisfied. Even Krieg-tic-tac-toe (a blind version of tic-tac-toe) can be modelled by a structure where actions correspond to *trying to mark a cell*, instead of *marking a cell*. The action of trying to mark is always allowed, but the result of its execution is different depending on the actual situation.

We provide examples of models in the next sections (see Fig. 1, 2, 3 and 4).

In Sec. IV-B, we prove some properties about the relation between agent intentions and knowledge in games. To achieve this, we need a logical language to express such properties. Therefore, we follow [18], [19] and define a kind of alternating-time epistemic logic (ATEL). For our purposes though, we do not need the full expressiveness of ATEL. We only need what could be called the "next-fragment" of ATEL. This is the basic language of our formalism.

Definition 3 (Basic Language): Let a game signature (P, N, A) be given. The language $\mathcal{L}_0$ is the set of formulas ϕ defined by the following grammar in BNF:

$$\phi ::= \top \mid p \mid \neg\phi \mid \phi \vee \phi \mid \mathrm{K}_i\phi \mid \mathrm{EX}\phi$$

where p ranges over P and i ranges over N. We use the common abbreviations for the symbols $\bot$, $\wedge$, $\to$, $\leftrightarrow$, and $\mathrm{AX}\phi$ abbreviates $\neg\mathrm{EX}\neg\phi$.

A formula of the form $\mathrm{K}_i\phi$ is read 'agent i knows that ϕ is true'. A formula of the form $\mathrm{EX}\phi$ is read 'ϕ is true after the execution of some joint action α'. Consequently, a formula of the form $\mathrm{AX}\phi$ is read 'ϕ is true after the execution of any executable joint action α'. Operators EX and AX have been inspired by operators 'all' (A) 'exists' (E) and 'next' (X) of computation tree logic (CTL) [20]. They also have analogous semantics.

Definition 4 (Basic Satisfaction Relation): The satisfaction relation $\models$ between pointed models and formulas in $\mathcal{L}_0$ is defined recursively, as follows:

$$\begin{array}{lll} \mathcal{M}, w \models \top & & \text{(always)} \\ \mathcal{M}, w \models p & \text{iff} & \pi(w)(p) = 1 \\ \mathcal{M}, w \models \neg\phi & \text{iff} & \mathcal{M}, w \not\models \phi \\ \mathcal{M}, w \models \phi \vee \psi & \text{iff} & \mathcal{M}, w \models \phi \text{ or } \mathcal{M}, w \models \psi \\ \mathcal{M}, w \models \mathrm{K}_i\phi & \text{iff} & \text{for all } w' \in [w]_i \text{ we have } \mathcal{M}, w' \models \phi \\ \mathcal{M}, w \models \mathrm{EX}\phi & \text{iff} & \text{there is } \alpha \in D(w) \text{ s.t. } \mathcal{M}, \delta(w,\alpha) \models \phi \end{array}$$

When $\mathcal{M}$ is clear from the context, we sometimes write $w \models \phi$ instead of $(\mathcal{M}, w) \models \phi$. We also use $[\![\phi]\!]_{\mathcal{M}}$ to denote the extension of ϕ in $\mathcal{M}$, i.e., $[\![\phi]\!]_{\mathcal{M}} = \{w \mid \mathcal{M}, w \models \phi\}$.

Let $\mathcal{M}$ be the model in Fig. 2. We have, e.g., $w_1 \models \neg(p \vee q)$, meaning that both p and q are false at w_1, $w_0 \models \mathrm{AX}\neg(p \vee q)$, meaning that both p and q are false after the execution of any action at w_0, and $w_0 \models \neg\mathrm{K}_i\mathrm{AX}\neg(p \vee q)$ means that i does not know that both p and q are false after the execution of any action at w_0.

Definition 5 (Validity and Satisfiability): A formula $\phi \in \mathcal{L}_0$ is valid (noted $\models \phi$) iff $(\mathcal{M}, w) \models \phi$, for all pointed models $(\mathcal{M}, w)$. A formula $\phi \in \mathcal{L}_0$ is satisfiable iff $\not\models \neg\phi$.

Examples of valid formulas include the axioms of modal logic S5 for the knowledge operator: $\mathrm{K}_i(\phi \to \psi) \to (\mathrm{K}_i\phi \to \mathrm{K}_i\psi)$, $\mathrm{K}_i\phi \to \phi$, and $\neg\mathrm{K}_i\phi \to \mathrm{K}_i\neg\mathrm{K}_i\phi$, as well as modal logic axiom K for the dual of the temporal operator: $\mathrm{AX}(\phi \to \psi) \to (\mathrm{AX}\phi \to \mathrm{AX}\psi)$.

The logic proposed here is similar to several other logics proposed to model agents' intentions, namely, all those mentioned in the beginning of the section. For instance, Khan and Lespérance's approach [15] also defines transition function for actions and accessibility relations for knowledge. Being a version of situation calculus (thus a fragment of FOL), their formalism is more expressive than ours, but has less attractive computational complexity. The parametrized-time action logic (PAL) by van Zee et al. [14] is a modal logic, like ours, but it is a single-agent formalism, which is obviously inadequate to represent games. In addition, PAL represents agent's beliefs with belief databases and does not have the knowledge (or rather belief) operator in the language.

III. Intention

Suppose that one wants to know whether, at a given state w of a game, agent i *acts with the intention to achieve ϕ after k steps*. We propose that the answer to this question is *yes* when the agent decides to execute an action a such that the following four conditions are true.

Conditions of Intention:

1) From the agent's perspective, action a is among the best ones to achieve ϕ. We consider this as the most basic principle of intention. We believe that one can conclude that an agent acts with the intention to achieve ϕ only when the agent decides to execute one of the actions that the agent thinks have the best chance to achieve ϕ.
2) From the agent's perspective, action a possibly leads to ϕ. Indeed, if the opposite is known to be true (i.e., a is known to lead to $\neg\phi$) then clearly, the agent is not acting with the intention to achieve ϕ.
3) From the agent's perspective, there is an option that may prevent ϕ, i.e., there is a different action b that could lead to $\neg\phi$. If there is no such alternative, then it is not possible to conclude that the agent has the intention to achieve ϕ, because ϕ would be unavoidable. This condition is inspired by some theories of action (e.g., [21]), which stipulate that choice (which relates to intention) only exists when it is possible to "do otherwise".
4) From the agent's perspective, action a is not one of the best to achieve $\neg\phi$. If this is not true, the agent's decision is ambiguous with respect to ϕ and $\neg\phi$. We believe that in such a case, it is not possible to conclude whether the agent acts with the intention to achieve ϕ or $\neg\phi$.

In the sequel, we describe a method that implements these conditions. To do so, we give a formal definition for the 'best actions to achieve ϕ'. It is done by calculating what we call here 'actions success rates'. Intuitively, it corresponds to the number of times an individual action a achieves ϕ over the number of all possible outcomes of a. Albeit joint actions are deterministic, individual actions have several possible outcomes (i.e., $\delta_i(w, a)$ is a set), because the actual outcome depends on the actions executed at the same time by the other agents of the scenario. Now, assuming that $\mathrm{sr} : N \times W \times A \times \mathbb{N} \times \mathcal{L}_0 \to [0, 1]$ is such a success rate function, agent intentions can be formally defined as follows.

Definition 6 (Intention): Let $\mathrm{maxsr}_i(w, k, \phi)$ denote the maximum action success rate to achieve ϕ after k steps, i.e., the maximum of the set $\{0\} \cup \{\mathrm{sr}_i(w, a, k, \phi) \mid a \in d_i(w)\}$. Agent i acts with the intention to achieve ϕ after k steps from state w if and only if i executes an action $a \in A$ for which all conditions below are true:

$$\mathrm{sr}_i(w, a, k, \phi) \geq \mathrm{maxsr}_i(w, k, \phi) \tag{1}$$

$$\exists b \in d_i(w) \text{ s.t. } \mathrm{sr}_i(w, b, k, \phi) < \mathrm{maxsr}_i(w, k, \phi) \tag{2}$$

$$\begin{array}{l} \mathrm{sr}_i(w, a, k, \neg\phi) = 0 \text{ or} \\ \mathrm{sr}_i(w, a, k, \neg\phi) < \mathrm{maxsr}_i(w, k, \neg\phi) \end{array} \tag{3}$$

Equation 1 (resp. 2 and 3) in Def. 6 formalizes Cond. 1 (ref. 3 and 4) saw earlier. Also note that Eq. 1 and 2 together imply $\mathrm{sr}_i(w, a, k, \phi) > 0$, which corresponds to Cond. 2. Therefore, Def. 6 formalizes the four conditions of intention introduced previously.

The formal definition of the action success rate function sr is given in Table I. Because this is a long and complicated definition, the next three subsections are entirely devoted to explain the intuitions underlying it. We start with simple scenarios and then generalize them for more complicated ones step by step.

A. Perfect Information Single-agent Scenarios

Let us first assume a perfect information single-agent scenario. That is, assume models where N and $[w]_i$ are singletons.

TABLE I
SUCCESS RATE FUNCTION DEFINITION

Action success rate: $\mathrm{sr}_i(w,a,k,\phi) = \mathrm{avg}_{w' \in [w]_i}\{\mathrm{sr}'_i(w',a,k,\phi)\}$
where:

$$\mathrm{sr}'_i(w,a,k,\phi) = \begin{cases} \underset{\alpha \in \mathrm{bjac}_i(w,a)}{\mathrm{avg}} \{\mathrm{v}_i(\delta(w,\alpha),k-1,\phi)\}, & \text{if } a \in d_i(w) \\ 0, & \text{otherwise} \end{cases}$$

State value:

$$\mathrm{v}_i(w,k,\phi) = \begin{cases} \mathrm{pr}([w]_i,\phi), & \text{if } k=0 \\ \max_{a \in A}\{\mathrm{avg}_{w' \in [w]_i}\{\mathrm{sr}_i(w',a,k,\phi)\}\}, & \text{if } k>0 \end{cases}$$

Presence ratio: $\mathrm{pr}([w]_i,\phi) = \frac{|[w]_i \cap [\![\phi]\!]_{\mathcal{M}}|}{|[w]_i|}$

Best joint actions: $\mathrm{bjac}_i(w,a) = \{a\} \times \prod_{j \in N \setminus \{i\}} \mathrm{bac}_j(w)$

Best action: $\mathrm{bac}_i(w) = \mathrm{argmax}_{a \in d_i(w)}\{E[u_i(w,a)]\}$

Expected action utility: $E[u_i(w,a)] = \mathrm{avg}_{w' \in [w]_i}\{u_i(w',a)\}$

In such cases, the action success rate function sr is defined as follows:

$$\mathrm{sr}_i(w,a,k,\phi) = \begin{cases} \mathrm{v}_i(\delta_i(w,a),k-1,\phi), & \text{if } a \in d_i(w) \\ 0, & \text{otherwise} \end{cases}$$

$$\mathrm{v}_i(w,k,\phi) = \begin{cases} 0, & \text{if } k=0 \text{ and } w \notin [\![\phi]\!]_{\mathcal{M}} \\ 1, & \text{if } k=0 \text{ and } w \in [\![\phi]\!]_{\mathcal{M}} \\ \max\limits_{a \in A}\{\mathrm{sr}_i(w,a,k,\phi)\}, & \text{if } k>0 \end{cases}$$

In words, the success rate of an action a equals the value of the state $\delta_i(w,a)$, which is the maximum success rate of all i's actions at that state. That is, the success rate of a is either 0 or 1, according to whether $\delta_i(a,w)$ lays on a path to ϕ after k steps. The values are always 0 or 1 because in this kind of scenario the future is completely foreseeable by the agent, which means that the agent only needs to calculate the outcomes of actions to decide what to do.

Example 1: In the model depicted in Fig. 1, we have:

$$\mathrm{sr}_i(w_0,a,1,\neg p) = \mathrm{v}_i(w_1,0,\neg p) = 1$$
$$\mathrm{sr}_i(w_0,b,1,\neg p) = \mathrm{v}_i(w_2,0,\neg p) = 0$$
$$\mathrm{sr}_i(w_0,a,1,p) = \mathrm{v}_i(w_1,0,p) = 0$$
$$\mathrm{sr}_i(w_0,b,1,p) = \mathrm{v}_i(w_2,0,p) = 1$$

If i executes a at w_0, then i acts with the intention to achieve $\neg p$ after 1 step. Indeed, all conditions are satisfied: a is the best action to achieve $\neg p$, there is another action that could achieve p and a is not one of the best actions to achieve p.

The reader may also verify that, if i executes a, then i also acts with the intention to achieve $\top$ after 2 steps. This means that the agent acts with the intention to make the game lasts for one more step. Again, this is true because action a is the best one to achieve $\top$ after 2 steps, there is another action which does not have that result and a is not one of the best to achieve $\neg\top$ after 2 steps.

We cannot conclude that i acts with the intention to achieve p after 2 steps nor that the agent acts with the intention to achieve $\neg p$ after 2 steps. The reason is that a is the best action

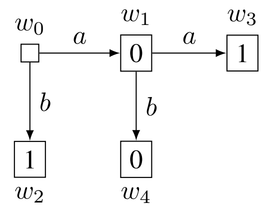

Fig. 1. The model of a perfect information single-agent game. Rectangles represent states. The numbers are the truth values of p. The valuation at the empty rectangle is irrelevant for the example. Arrows represent actions. There are three actions available: a, b and c. Action c is illegal at all states.

to achieve both results after 2 steps. Finally, the agent has no intention after 3 steps. That is, there is no formula $\phi \in \mathcal{L}_0$ for which all the conditions are true, when considering 3 steps.

B. Imperfect Information Single-agent Scenarios

Now, we drop the assumption that sets $[w]_i$ are singletons. In this case, function sr is generalized as follows:

$$\mathrm{sr}_i(w,a,k,\phi) = \underset{w' \in [w]_i}{\mathrm{avg}}\{\mathrm{sr}'_i(w',a,k,\phi)\}$$

$$\mathrm{sr}'_i(w,a,k,\phi) = \begin{cases} \mathrm{v}_i(\delta_i(w,a),k-1,\phi), & \text{if } a \in d_i(w) \\ 0, & \text{otherwise} \end{cases}$$

and the state value function v is the one in Table I.

Function v no longer calculates the value of a state, but the value of a set of indistinguishable states. When $k=0$, the value of state w is equal to the presence ratio of ϕ in $[w]_i$, which is calculated with the 'presence ratio' function pr. The latter calculates the number of different indistinguishable states where ϕ is true over the total number of different indistinguishable states for agent i.[1] The idea of using this ratio to calculate intention was first proposed in [23]. The intuitive idea is that the agent ascribes equal probability to each possible state indistinguishable from the current one. The action success rate is calculated as the average of the success rates in each state of $[w]_i$. Actions success rates are not defined for $k=0$ because it does not make sense to try to achieve ϕ after 0 steps with an action (whose execution necessarily consumes a step).

Example 2 (Coffee and Milk): Assume an academic department with a cafeteria and a kitchen. In the cafeteria, there is a coffee machine which delivers coffee (p_1) and milk (p_2), but that works only 50% of the time. In the kitchen there is, $1/3$ of the time, a vacuum bottle with coffee and no milk; $1/3$ of the time a bottle with milk and no coffee, and no beverage otherwise. The agent can go to the cafeteria (action a) or to the kitchen (action b). This scenario is depicted in Fig. 2.

[1]By 'different states', we mean non-bisimilar states. By the invariance theorem of modal logic (see e.g., [22]) states w and w' are bisimilar if and only if, for all $\phi \in \mathcal{L}_0$, $w \vDash \phi$ iff $w' \vDash \phi$. We need this assumption because function sr is sensible to the number of states in the model, and nothing forbids a model to have duplicated states. For example, if the root state is duplicated (hence, indistinguishable from the original root state) it is still a valid model. In such a case, the entire game tree would be duplicated and the calculations would be affected. Bisimilar states can be eliminated from the model by a process called bisimulation contraction [22, p. 14].

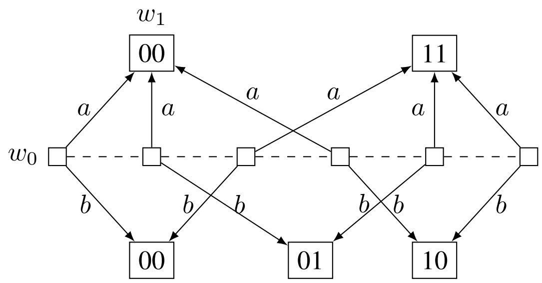

Fig. 2. The model of Example 2 (Coffee and Milk). At each state, the truth values of p_1 and p_2, respectively. Dashed lines represent relation $\sim_i$.

The values below show that, if i goes to the cafeteria (action a), then she acts with the intention to have coffee **and** milk:

$$\mathrm{sr}_i(w_0, a, 1, p_1 \wedge p_2) = 1/2$$
$$\mathrm{sr}_i(w_0, b, 1, p_1 \wedge p_2) = 0$$
$$\mathrm{sr}_i(w_0, a, 1, \neg(p_1 \wedge p_2)) = 1/2$$
$$\mathrm{sr}_i(w_0, b, 1, \neg(p_1 \wedge p_2)) = 1$$

whereas, if i goes to the kitchen (action b), then she acts with the intention to have coffee **or** milk:

$$\mathrm{sr}_i(w_0, a, 1, p_1 \vee p_2) = 1/2$$
$$\mathrm{sr}_i(w_0, b, 1, p_1 \vee p_2) = 2/3$$
$$\mathrm{sr}_i(w_0, a, 1, \neg(p_1 \vee p_2)) = 1/2$$
$$\mathrm{sr}_i(w_0, b, 1, \neg(p_1 \vee p_2)) = 1/3$$

The example above shows that, in general, intention to achieve $p_1 \wedge p_2$ does not imply intention to achieve $p_1 \vee p_2$. In fact, this example reveals something even less intuitive than that. Note that, if i executes b, then i acts with the intention to achieve $p_1 \vee p_2$ and also acts with the intention to achieve $\neg p_1$. However, the agent should execute a to act with the intention to achieve p_2. This means that intention to achieve $p_1 \vee p_2$ and intention to achieve $\neg p_1$ does not imply intention to achieve p_2. Another way of saying that is: intention does not satisfy modal logic axiom K. We analyse more such properties satisfied (or not) by intentions in Sec. IV-B.

C. Multiagent Scenarios

In imperfect information multiagent scenarios, we drop both the assumptions that N and $[w]_i$ are singletons. The success rate function is generalized once again. It is now the one presented in Table I. Each action executed by agent i can lead to multiple different states, because of the different actions that the other agents can execute at the same time. This is why we use a 'best response' function bjac in the definition of the success rate function sr. Consequently, in these scenarios actions success rates are not calculated based on the entire game tree, but only on the paths that agents would actually take in the game when playing rationally.

Example 3 (Tic-tac-toe): We model the well known game Tic-tac-toe. The game signature consists of:

$$P = \{\mathrm{cell}(l, c, i) \mid l \in \{0, 1, 2\} \text{ and } i \in N\}$$
$$N = \{\mathrm{X}, \mathrm{O}\}$$
$$A = \{\mathrm{noop}\} \cup \{\mathrm{mark}(l, c) \mid l, c \in \{0, 1, 2\}\}$$

Fig. 3 depicts a small part of the model (the entire tree has approximately $5! = 120$ nodes) where the initial board is already partially marked and it is X's turn to play. Because it is a zero-sum game, utilities can be calculated using the minimax algorithm.

Agent X can win the game in various ways: if she marks cell $(0, 0)$ (which leads to w_1), she can complete the first row or the first diagonal, and if she marks cell $(0, 1)$, she can complete either the first row or the second column. However, if she marks cell $(1, 0)$, then she intends **not** to win the game. To see why, first consider the following goal formulas:

$$\phi_0 = \mathrm{cell}(0, 0, \mathrm{X}) \wedge \mathrm{cell}(0, 1, \mathrm{X}) \wedge \mathrm{cell}(0, 2, \mathrm{X})$$
$$\phi_1 = \mathrm{cell}(0, 0, \mathrm{X}) \wedge \mathrm{cell}(1, 1, \mathrm{X}) \wedge \mathrm{cell}(2, 2, \mathrm{X})$$
$$\phi_2 = \mathrm{cell}(0, 1, \mathrm{X}) \wedge \mathrm{cell}(1, 1, \mathrm{X}) \wedge \mathrm{cell}(2, 1, \mathrm{X})$$
$$\psi = \phi_0 \vee \phi_1 \vee \phi_2$$

Formula ϕ_0 means that X completes the first row on the board, ϕ_1 means that X completes the first diagonal, ϕ_2 means that X completes the middle column, and ψ means that X wins the game in Fig. 3. We have:

$$\mathrm{sr}_{\mathrm{X}}(w_0, \mathrm{mark}(1, 0), 5, \neg\psi) = 1 \quad (4)$$
$$\mathrm{sr}_{\mathrm{X}}(w_0, \mathrm{mark}(1, 0), 5, \psi) = 0 \quad (5)$$
$$\mathrm{sr}_{\mathrm{X}}(w_0, \mathrm{mark}(0, 0), 5, \neg\psi) = 1/2 \quad (6)$$

This means that, by marking cell $(1, 0)$, X acts with the intention **not** to win the game after 5 steps. Indeed, (4) means that, after marking $(1, 0)$, X is certain to obtain a non-winning state (i.e., a state satisfying $\neg\psi$) after 5 steps. Equation (5) shows that, after marking $(1, 0)$, X never obtains a winning state after 5 steps. Therefore, marking $(1, 0)$ is not one of the best actions to achieve ψ after 5 steps. Equation (6) means that, after marking $(0, 0)$, X obtains a non-winning state after 5 steps in 50% of the cases. Therefore, there is a non-best action for $\neg\psi$.

It may seem puzzling that action $\mathrm{mark}(0, 0)$, which is part of a winning strategy for X, does not lead to a winning state in all cases. Te reason is that, when querying for $\neg\psi$, we assume that X is aiming at satisfying this formula and we disregard the utility function u_{X}. But note that the definition in Table I uses the utility function u_{O} for the opponent agent O, which means that we assume that O acts to maximize her utility.

It is interesting to see that not all moves reveal agent's intentions. For example, consider the following success rates:

$$\mathrm{sr}_{\mathrm{X}}(w_0, \mathrm{mark}(0, 0), 3, \phi_0) = 3/4$$
$$\mathrm{sr}_{\mathrm{X}}(w_0, \mathrm{mark}(0, 0), 3, \neg\phi_0) = 1$$
$$\mathrm{sr}_{\mathrm{X}}(w_0, \mathrm{mark}(1, 0), 3, \phi_0) = 0$$

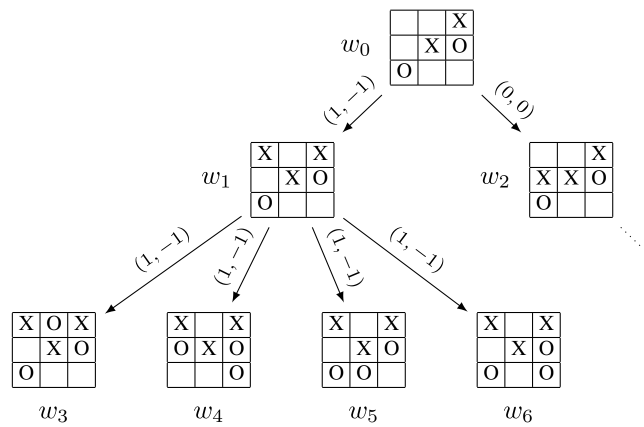

Fig. 3. The model of Example 3 (Tic-tac-toe). The numbers on the arrows are the utilities for agents i and j, respectively.

The first value above indicates that action $\mathrm{mark}(0,0)$ has a success rate of $3/4$ to achieve ϕ_0 after 3 steps for X. Indeed, in Fig. 3 we can see that, if X aims at completing the first row after 3 steps, then X can achieve it from states w_4, w_5 and w_6, but not from w_3. However, this does not show that X has the intention to achieve ϕ_0 after 3 steps because, even though action $\mathrm{mark}(0,0)$ is among the best ones to achieve ϕ_0 after 3 steps and there is at least one legal action, namely $\mathrm{mark}(1,0)$, that is not among of the best ones to achieve that, we have that (which may seem surprising) $\mathrm{mark}(0,0)$ is among the best actions to achieve $\neg\phi_0$ as well. What prevents us from being able to infer X's intention is that X can still deviate from that objective in her next move. In other words, it is too soon to conclude that X is aiming at completing the first row after 3 steps. Maybe X is aiming at completing the diagonal, or maybe she is planning to mark cell $(1,0)$ next, which means that she does not want to win the game. It is just not possible to know by analyzing only one action. Also note that $\mathrm{mark}(0,1)$ has the same success rate values, thus the same reasoning applies.

IV. A Logic of Intention

A. Intention Operator

Latter, we show some properties satisfied by our definition of intention. To do this, we extend the language with an intention operator.

Definition 7: Let a game signature (P, N, A) be given. The language $\mathcal{L}$ is the set of formulas χ defined by the following grammar in BNF:

$$\chi ::= \top \mid p \mid \neg\chi \mid \chi \vee \chi \mid \mathrm{K}_i\chi \mid \mathrm{EX}\chi \mid \mathrm{I}_i^a\mathrm{X}^k\phi$$

where p ranges over P, i ranges over N, a ranges over A, k ranges over $\mathbb{N}^+$ and ϕ ranges over $\mathcal{L}_0$. We sometimes use $\mathrm{I}_i^a\mathrm{X}\phi$ to abbreviate $\mathrm{I}_i^a\mathrm{X}^1\phi$.

In the intention operator (as the reader has probably guessed), 'I' stands for 'intention', whereas 'X' remembers us that this is an intention operator towards the future. The intended meaning of formulas of the form $\mathrm{I}_i^a\mathrm{X}^k\phi$ is 'when i executes a she intends to achieve ϕ after k steps'.

Formula ϕ ranges over $\mathcal{L}_0$ in Def. 7. Consequently, $\mathcal{L}$ does not admit nesting of the intention operator. Thus, e.g., $\mathrm{I}_i^a\mathrm{X}(p_1 \wedge \mathrm{I}_i^a\mathrm{X}p_2)$ is **not** a well-formed formula. Also note that $\mathrm{I}_i^a\mathrm{XX}p$ is **not** well-formed either (because of the two symbols 'X' together).

Examples of well-formed formulas are: $\mathrm{I}_i^a\mathrm{X}^2p$ (meaning 'when i executes a she intends to achieve p after 2 steps'), $\mathrm{I}_i^a\mathrm{X}(\mathrm{K}_ip \vee \neg\mathrm{K}_ip)$ (when i executes a she intends to know whether p is true), $\mathrm{K}_i\mathrm{I}_j^a\mathrm{X}p$ (i knows that, when j executes a she intends to achieve p after 1 step) and $\mathrm{I}_i^a\mathrm{X}p_1 \rightarrow \mathrm{I}_i^a\mathrm{X}p_2$ (when i executes a she intends to achieve p_1 after 1 step implies when i executes a she intends to achieve p_2 after 1 step).

Definition 8 (Satisfaction Relation): The satisfaction relation $\vDash$ between formulas in $\mathcal{L}$ and pointed models $(\mathcal{M}, w)$ is the same as in Def. 4 for the Boolean, knowledge and temporal operators plus:

$$\mathcal{M}, w \vDash \mathrm{I}_i^a\mathrm{X}^k\phi \quad \text{iff} \quad a \in \mathrm{B}_i(w,k,\phi) \setminus \mathrm{B}_i(w,k,\neg\phi) \text{ and } \mathrm{B}_i(w,k,\phi) \neq d_i(w)$$

where:

$$\mathrm{B}_i(w,k,\phi) = \{a \in A \mid \mathrm{sr}_i(w,a,k,\phi) > 0 \text{ and } \mathrm{sr}_i(w,a,k,\phi) = \mathrm{maxsr}_i(w,k,\phi)\}$$

The semantics of the operators already in $\mathcal{L}_0$ are the same. For the intention operator, we have that, $\mathrm{I}_i^a\mathrm{X}^k\phi$ is true if and only if action a is one of the best actions to achieve ϕ after k steps, is not one of the best actions to achieve $\neg\phi$ after k steps, and the set of best actions to achieve ϕ after k steps is not the entire set of executable (legal) actions. This last condition implies that there is a different action b that is one of the best to achieve $\neg\phi$ after k steps. Therefore, this semantics implements Def. 6.

For example, let $\mathcal{M}$ be the model in Fig. 3, and let $a = \mathrm{mark}(1,0)$ and $b = \mathrm{mark}(0,0)$. We have $w_0 \vDash \mathrm{I}_i^a\mathrm{X}^5\neg\psi$ and $w_0 \vDash \neg\mathrm{I}_i^b\mathrm{X}^3\phi_0$.

B. Properties of the Intention Operator

We now aim at showing that operator $\mathrm{I}_i^a\mathrm{X}^k$ is indeed an intention operator. In order to do so, we show that it satisfies some desired properties, the most important being the one called *intention-belief consistency*. Indeed, this property seems to be shared among all theories of intention. For instance, Bratman's Asymmetry Thesis states that:

> "An intention to A normally provides the agent with support for a belief that he will A. But there need be no irrationality in intending to A and yet still not believing one will. In contrast, there will be irrationality in intending to A and believing one will not A; for there is a defensible demand that one's intentions be consistent with one's beliefs." [4, p.38].

The more recent Shoham's database perspective approach to intention [5], [14] stipulates that an agent that intends to take an action cannot believe that its preconditions do not hold, and also believes that the post-conditions of the action hold after its execution.

The view that beliefs and intentions must be consistent with each other is also adopted by several logical approaches to intention [2], [6], [10]. In BDI logic [1], this has been interpreted as the following three principles:

1) $\models \mathsf{INT}(\phi) \rightarrow \neg\mathsf{BEL}(\neg\phi)$ (intention-belief consistency)
2) $\not\models \mathsf{INT}(\phi) \rightarrow \mathsf{BEL}(\phi)$ (intention-belief incompleteness)
3) $\not\models \mathsf{BEL}(\phi) \rightarrow \mathsf{INT}(\phi)$ (belief-intention incompleteness)

Cohen and Levesque [10] presented an example usually called the 'dentist example':

> "I intend to get my teeth fixed, and I know that it implies experiencing pain, but I do not intend to experience pain."

They use that example to defend a principle which can be formalized as follows:

4) $\not\models (\mathsf{INT}(\phi) \wedge \mathsf{BEL}\Box(\phi \rightarrow \psi)) \rightarrow \mathsf{INT}(\psi)$ (side-effect)

Roy [6] defends the following four principles in his approach to intention:

5) $\models \phi$ implies $\models \mathrm{I}_i\phi$ (necessitation)
6) $\models \mathrm{I}_i(\phi \rightarrow \psi) \rightarrow (\mathrm{I}_i\phi \rightarrow \mathrm{I}_i\psi)$ (axiom K)
7) $\models \mathrm{I}_i\phi \rightarrow \neg\mathrm{I}_i\neg\phi$ (seriality)
8) $\models \mathrm{I}_i\phi \rightarrow \mathrm{K}_i\mathrm{I}_i\phi$ (interaction)

Khan and Lespérance [15] also show that their approach satisfies several properties. The ones that can be translated to our formalism are:

9) $\models \neg\mathrm{I}_i\mathrm{X}\bot$ (consistency)
10) $\models \mathrm{K}_i\mathrm{AX}\phi \rightarrow \mathrm{I}_i\mathrm{X}\phi$ (realism)
11) $\models \mathrm{I}_i\mathrm{X}\phi \rightarrow \neg\mathrm{K}_i\mathrm{AX}\neg\phi$ (int.-knowledge consistency)

There is no belief operator in our logic. The way we can show that our approach respects intention-belief consistency is indirect. We appeal to a relation between knowledge and belief that is widely accepted: 'what is known is also believed'. This amounts to the validity of $\mathrm{K}_i\phi \rightarrow \mathsf{BEL}_i\phi$. This means that, if Principle 1 were valid in a logic with a belief and a knowledge operator, then the formula $\mathrm{I}_i^a\mathrm{X}\phi \rightarrow \neg\mathrm{K}_i\mathrm{AXK}_i\neg\phi$ should also be valid. We call this principle (as in [15]) *intention-knowledge consistency*. Note the use of $\mathrm{K}_i\mathrm{AXK}_i$ instead of just K_i. This is necessary in our setting because, as said earlier, when we write $\mathrm{I}_i^a\mathrm{X}\phi$, we mean that i intends to achieve ϕ in the *next* state. This will be a requirement for all the principles we address in the sequel. We prove several principles for our intention operator in Prop. 2 below, which uses the following lemma.

Lemma 1: $\mathcal{M}, w \models \mathrm{K}_i\mathrm{AXK}_i\phi$ implies $\mathrm{B}_i(w, 1, \phi) = d_i(w)$.

Proof: By definition, $(\mathcal{M}, w) \models \mathrm{K}_i\mathrm{AXK}_i\phi$ iff, for all $w' \in [w]_i$, all $d_i(w)$ and all $w'' \in [\delta(\alpha, w')]_i$, we have $(\mathcal{M}, w'') \models \phi$. This means that $\mathrm{pr}([w'']_i, \phi) = 1$ (for all w''). Then we have $\mathrm{v}_i(w'', 0, \phi) = 1$ (for all w''), and then $\mathrm{v}_i(\delta(w', \alpha), 1, \phi) = 1$ (for all w' and $\alpha \in D(w')$). This implies, for all $a \in d_i(w)$, $\mathrm{sr}_i(w, a, 1, \phi) = 1$. Therefore, $a \in \mathrm{B}_i(w, 1, \phi)$, for all $a \in d_i(w)$. ■

In what follows, $(\mathrm{AX})^k$ (resp. $(\mathrm{AXK}_i)^k$) denotes a sequence of k operators, e.g., $(\mathrm{AX})^3\phi = \mathrm{AXAXAX}\phi$, and $(\mathrm{AXK}_i)^2 = \mathrm{AXK}_i\mathrm{AXK}_i\phi$. To avoid confusion, the superscript k is never used in this way with the intention operator.

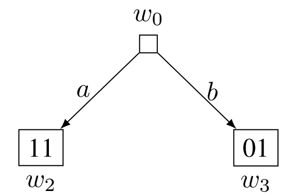

Fig. 4. A counter-model for the formula in Prop. 2.5. The numbers are the truth values of p_1 and p_2, respectively.

Proposition 2:

1) $\models \mathrm{I}_i^a\mathrm{X}\phi \rightarrow \neg\mathrm{K}_i\mathrm{AXK}_i\neg\phi$ (int.-knowledge consist.)
2) $\models \mathrm{I}_i^a\mathrm{X}^k\phi \rightarrow \neg\mathrm{K}_i(\mathrm{AXK}_i)^k\neg\phi$ (k-steps int.-knowledge consist.)
3) $\not\models \mathrm{I}_i^a\mathrm{X}\phi \rightarrow \mathrm{K}_i\mathrm{AXK}_i\phi$ (int.-knowledge incompl.)
4) $\not\models \mathrm{K}_i(\mathrm{AXK}_i)^k\phi \rightarrow \mathrm{I}_i^a\mathrm{X}^k\phi$ (k-steps knowledge-int. incompl.)
5) $\not\models (\mathrm{I}_i^a\mathrm{X}^k\phi \wedge \mathrm{K}_i(\mathrm{AX})^k(\phi \rightarrow \psi)) \rightarrow \mathrm{I}_i^a\mathrm{X}\psi$ (side-effect)
6) $\models \neg\mathrm{I}_i^a\mathrm{X}^k\bot$ (consistency)
7) $\models \mathrm{I}_i^a\mathrm{X}^k\phi \rightarrow \neg\mathrm{I}_i^a\mathrm{X}^k\neg\phi$ (seriality)
8) $\models \mathrm{I}_i^a\mathrm{X}^k\phi \rightarrow \mathrm{K}_i\mathrm{I}_i^a\mathrm{X}^k\phi$ (interaction)
9) From $\phi \leftrightarrow \psi$ infer $\mathrm{I}_i^a\mathrm{X}^k\phi \leftrightarrow \mathrm{I}_i^a\mathrm{X}^k\psi$ (extensionality)

Proof:

1) Assume the opposite, i.e., there is $(\mathcal{M}, w)$ that satisfies $\mathrm{I}_i^a\mathrm{X}\phi \wedge \mathrm{K}_i\mathrm{AXK}_i\neg\phi$. This means that $\mathrm{B}_i(w, 1, \neg\phi) = d_i(w)$ (by Lemma 1). The latter implies $(\mathcal{M}, w) \not\models \mathrm{I}_i^a\mathrm{X}\phi$ (by Def. 8). But this contradicts the assumption. Therefore, we have that $\mathrm{I}_i^i\mathrm{X}\phi \rightarrow \neg\mathrm{K}_i\mathrm{AXK}_i\neg\phi$ is valid.
2) This can be proved by iterating the arguments in the proofs of Lemma 1 and Prop. 2.1 k times.
3) Let $\mathcal{M}$ be the model in Fig. 2. We have that $(\mathcal{M}, w_0) \models \mathrm{I}_i^a\mathrm{X}p$, because $a \in \mathrm{B}_i(w_0, 1, p) \setminus \mathrm{B}_i(w_0, 1, \neg p)$ and $\mathrm{B}_i(w_0, 1, p) \neq d_i(w_0)$. However, we clearly have that $(\mathcal{M}, w_0) \not\models \mathrm{K}_i\mathrm{AXK}_ip$, because $\delta(w_0, a) = w_1$ and $(\mathcal{M}, w_1) \not\models p$.
4) We have $(\mathcal{M}, w) \models \mathrm{K}_i(\mathrm{AXK}_i)^k\phi$ implies $\mathrm{B}_i(a, k, \phi) = d_i(w)$ (by Lemma 1 iterated k times). Therefore, $(\mathcal{M}, w) \not\models \mathrm{I}_i^a\mathrm{X}^k\phi$.
5) We show it for $k = 1$. Let $\mathcal{M}$ be the model in Fig. 4. We have that $(\mathcal{M}, w_0) \models \mathrm{K}_i\mathrm{AX}(p_1 \rightarrow p_2)$, because p_2 is true in both w_2 and w_3. We also have $\mathrm{B}_i(w_0, 1, p_1) = \{a\}$ and $\mathrm{B}_i(w_0, 1, \neg p_1) = \{b\}$. Thus, $(\mathcal{M}, w_0) \models \mathrm{I}_i^a\mathrm{X}^1p_1$. However, because $\mathrm{B}_i(w_0, 1, p_2) = d_i(w_0)$, then we have $(\mathcal{M}, w_0) \not\models \mathrm{I}_i^a\mathrm{X}^1p_2$.
6) It is enough to see that $\mathrm{B}_i(w, k, \phi) = \mathrm{B}_i(w', k, \phi)$ for all $w' \in [w]_i$.
7) Immediate, because $\mathrm{B}_i(w, k, \bot) = \emptyset$, for all w.
8) Let $(\mathcal{M}, w) \models \mathrm{I}_i^a\mathrm{X}^k\phi$. Then $a \notin \mathrm{B}_i(\sigma, k, \neg\phi)$ (by Def. 6). Therefore, $(\mathcal{M}, w) \not\models \mathrm{I}_i^a\mathrm{X}^k\neg\phi$.
9) If $\models \phi \leftrightarrow \psi$ then $\mathrm{B}_i(w, k, \phi) = \mathrm{B}_i(w, k, \psi)$, for all w and k. Therefore, we have $\models \mathrm{I}_i^a\mathrm{X}^k\phi$ iff $\models \mathrm{I}_i^a\mathrm{X}^k(\psi)$. ■

Prop. 2.1, 2.3 and 2.4 correspond to Principles 1–3, respectively. Therefore, our intention operator satisfies all principles of the asymmetry thesis, but using knowledge instead of belief. Prop. 2.1 also corresponds to Principle 11. Prop. 2.5 corresponds to Principle 4. Prop. 2.6 corresponds to Principle 9. Prop. 2.7–2.8 correspond to Principles 7–8, respectively.

Three principles are not satisfied, namely, Principles 5, 6 and 10. Example 2 is a counter example for both Axiom K (Principle 6), Necessitation (Principle 5) (as well as Prop. 3.7 below). In addition, Principle 5 means 'agents always intend to achieve valid formulas'. This would contradict the third condition of intention (of Sec. III). Principle 10 contradicts the third principle of the asymmetry thesis. Therefore, it is preferable to not satisfy it. The following proposition lists these and some other interesting **non-validities**.

Proposition 3:

1) From ϕ we cannot infer $\mathrm{I}_i^a \mathrm{X}^k \phi$. (necessitation)
2) $\not\models \mathrm{I}_i^a \mathrm{X}^k(\phi \to \psi) \to (\mathrm{I}_i^a \mathrm{X}^k \phi \to \mathrm{I}_i^a \mathrm{X}^k \psi)$ (axiom K)
3) $\not\models \mathrm{I}_i^a \mathrm{X}^k \top$ (axiom N)
4) $\not\models \mathrm{I}_i^a \mathrm{X}^k(\phi \wedge \psi) \to (\mathrm{I}_i^a \mathrm{X}^k \phi \wedge \mathrm{I}_i^a \mathrm{X}^k \psi)$ (axiom M)
5) $\not\models (\mathrm{I}_i^a \mathrm{X}^k \phi \wedge \mathrm{I}_i^a \mathrm{X}^k \psi) \to \mathrm{I}_i^a \mathrm{X}^k(\phi \wedge \psi)$ (axiom C)
6) $\not\models (\mathrm{I}_i^a \mathrm{X}^k \phi \vee \mathrm{I}_i^a \mathrm{X}^k \psi) \to \mathrm{I}_i^a \mathrm{X}^k(\phi \vee \psi)$
7) $\not\models \mathrm{I}_i^a \mathrm{X}^k(\phi \wedge \psi) \to \mathrm{I}_i^a \mathrm{X}^k(\phi \vee \psi)$

V. Conclusion and Perspectives

In this work, we propose a method for checking agent intentions in games. Differently from most contributions, where intentions are directly provided in the model, ours infer agent intentions from their behaviour. We show that our operator indeed models intention as it satisfies the most important properties of intention, namely Bratman's asymmetry thesis as well as Cohen and Levesque's side-effect principle.

Maybe the most straightforward questions left open relate to the axiomatic system and computational complexity of the logic. Intention checking amounts to model checking formula $\mathrm{I}_i^a \mathrm{X}^k \phi$ in $(\mathcal{M}, w)$. It is decidable, because model checking in ATEL is decidable in polynomial time [18], [19] and the sets $\mathrm{B}_i(w, k, \phi)$, upon which intentions are defined, are finite. We are currently working on the adaptation of the existing model-checking techniques for this logic.

We saw in Example 3 that it is not possible to infer the intention of i with only one action. However, it would be possible if we could consider consecutive actions by i, e.g., marking $(0,0)$ and then marking $(0,1)$. It is not hard to generalize the success rate function to sequences of actions. Once this is done, a formula of the form $\mathrm{I}_i^{a_1 \cdots a_\ell} \mathrm{X}^k$ can be given a semantics that compares success rates of different sequences of ℓ actions by i and returns weather i acts with the intention to achieve ϕ after k steps. Again, we leave the details to future work.

We did not investigate "intentions about intentions". Our language does not permit formulas of the form $\mathrm{I}_i^a \mathrm{X} \mathrm{I}_j^b \mathrm{X} \phi$, which would mean 'when agent i executes a she intends to achieve, after 1 step, that, when j executes b she intends to achieve ϕ after 1 step'. This could mean delegation, persuasion, or coercion. Many questions arise. For instance, does intention about intention to ϕ imply intention to ϕ? What should be the agents intentions about their own intentions? Can it express intention change? Can an agent have the intention to change its intention? We plan to address these questions in the near future.

On the practical side, we are currently working on an implementation of our intention checking method for games in GDL-II [24]. The idea is to be able to check intentions of general game players.

References

[1] A. S. Rao and M. P. Georgeff, "Decision procedures for BDI logics," *J. of Logic and Computation*, vol. 8, no. 3, pp. 293–342, 1998.
[2] M. Wooldridge, *Reasoning about Rational Agents*. The MIT Press, 2000.
[3] A. Herzig, L. Emiliano, L. Perrussel, and Z. Xiao, "BDI logics for BDI architectures: Old problems, new perspectives," *Künstliche Intell.*, vol. 31, no. 1, pp. 73–83, 2017.
[4] M. E. Bratman, *Intention, Plans and Practical Reason*. Harvard University Press, 1987.
[5] Y. Shoham, "Logical theories of intention and the database perspective," *J. of Philosophical Logic*, vol. 38, pp. 633—647, 2009.
[6] O. Roy, "A dynamic-epistemic hybrid logic for intentions and information changes in strategic games," *Synthese*, vol. 171, no. 2, pp. 291–320, 2009.
[7] P. Galeazzi and E. Lorini, "Epistemic logic meets epistemic game theory: a comparison between multi-agent Kripke models and type spaces," *Synthese*, vol. 193, no. 7, pp. 2097–2127, 2016.
[8] W. Jamroga, W. van der Hoek, and M. Wooldridge, J., "Intentions and strategies in game-like scenarios," in *Proc. of EPIA*, ser. LNCS, vol. 3808. Springer, 2005, pp. 512–523.
[9] M. Eger, C. Martens, and M. A. Cordoba, "An intentional AI for hanabi," in *Proc. of CIG*. IEEE, 2017, pp. 68–75.
[10] P. Cohen and H. Levesque, "Intention is Choice with Commitment," *Artificial Intelligence*, vol. 42, pp. 213–261, 1990.
[11] J. Schrittwieser, I. Antonoglou, T. Hubert, K. Simonyan, L. Sifre, S. Schmitt, A. Guez, E. Lockhart, D. Hassabis, T. Graepel, T. Lillicrap, P., and D. Silver, "Mastering atari, go, chess and shogi by planning with a learned model," *CoRR*, 2019.
[12] M. J. Osborne and A. Rubinstein, *A Course in Game Theory*. The MIT Press, 2011.
[13] Y. Shoham and K. Layton-Brown, *Multiagent Systems*. Cambridge University Press, 2009.
[14] M. van Zee, D. Doder, L. van der Torre, M. Dastani, T. Icard, and E. Pacuit, "Intention as commitment toward time," *Artificial Intelligence*, vol. 283, p. 103270, 2020.
[15] S. M. Khan and Y. Lespérance, "A logical framework for prioritized goal change," in *Proc. of AAMAS*, 2010, pp. 283–290.
[16] P.-Y. Shobbens, "Alternating-time logic with imperfect recall," *Electronic Notes in Theoretical Computer Science*, vol. 85, no. 2, pp. 82–93, 2004.
[17] R. Alur, T. A. Henzinger, and O. Kupferman, "Alternating-time temporal logic," *J. of the ACM*, vol. 49, no. 5, pp. 672–713, 2002.
[18] T. Ågotnes, "Action and knowledge in alternating-time temporal logic," *Synthese*, vol. 149, pp. 375–407, 2006.
[19] W. van der Hoek and M. Wooldridge, "Cooperation, knowledge, and time: Alternating-time temporal epistemic logic and its applications," *Studia Logica*, vol. 75, pp. 125–153, 10 2003.
[20] E. M. Clarke and E. A. Emerson, "Design and synthesis of synchronization skeletons using branching time temporal logic," in *Proc. of Logics of Programs Workshop*, ser. LNCS, vol. 131, 1981, pp. 52–71.
[21] N. Belnap, M. Perloff, and M. Xu, *Facing the Future*. Oxford University Press, 2001.
[22] P. Blackburn, J. van Benthem, and F. Wolter, *Handbook of Modal Logic*, ser. Studies in Logic and Practical Reasoning. Elsevier, 2007, vol. 3.
[23] M. Eger, "Intentional agents for doxastic games," Ph.D. dissertation, 2018, North Caroline State University.
[24] M. Thielscher, "A general game description language for incomplete information games," in *Proc. of AAAI*, 2010.