Partial Information and Uniform Strategies

Hans van Ditmarsch and Sophia Knight

LORIA, CNRS – Université de Lorraine, France

Abstract. We present an alternating-time temporal epistemic logic with uniform strategies, interpreted in a novel way on transition systems for modelling situations in which agents with partial information interact to determine the way the system updates. This logic uATEL allows us to model what properties agents can enforce when they act according to strategies based on their knowledge. Apart from the usual memoryless strategies, we distinguish state-based memory, where agents recall the history of previous states, from perfect recall, where agents also recall their actions. We show that this makes a difference. Our logic includes three strategic operators for groups, representing the case where all the agents in the group cooperate actively, but do not share their knowledge, the case where some agents in the group may be passive, and the case where all the agents in the group share their knowledge. We include a detailed comparison to the literature on the subject.

1 Introduction

Linear temporal logic [15], Computation tree logic [6], and CTL* [7] are temporal logics for reasoning about distributed systems. LTL is concerned with infinite histories of states representing possible computations, CTL can reason about the branching structure of potential computations, and CTL* combines the expressive capabilities of both LTL and CTL. While the distributed systems that these logics model are implicitly understood to be systems consisting of many agents acting individually, there is no explicit mention of these agents or their actions in any of the above-mentioned temporal logics.

Alternating-time temporal logic [3] is a branching-time temporal logic for distributed systems where the effects of agents' actions are made explicit. In ATL, a formula of the form $\langle\!\langle \Gamma \rangle\!\rangle \bigcirc \varphi$, where Γ is a set of agents, $\bigcirc$ is the 'next' temporal modality, and φ is a state formula, signifies that the agents in the group Γ can cooperate to ensure that the formula φ is true in next state of the system, no matter what the agents who are not in Γ do. Thus, ATL provides a natural and interesting way of analyzing the properties of multi-agent systems, with the advantage of being able to analyse the effects of the actions of both specific agents and groups of agents.

Epistemic logic [13], on the other hand, is a different kind of modal logic which also models multi-agent systems, but only statically, without considering changes in these systems over time. Instead, epistemic logic is concerned with the agents' knowledge — another crucial aspect of multi-agent systems.

Thus, ATL is concerned with agents' abilities to control the outcomes of executions of dynamic multi-agent systems, while epistemic logic is concerned with agents' knowledge in static multi-agent systems. It is clear that combining these two focuses could yield a compelling and relevant logic for describing the interaction between agents' ability to act and their knowledge. Alternating-time temporal epistemic logic was proposed by [10] to combine knowledge and agency. Epistemic modalities are added to ATL, but the traditional semantics are used for both modalities. This approach allows some interesting applications, but since the original semantics are used for both modalities, the full interaction between the agents' knowledge and their actions is not captured. In particular, it is possible that an agent has different strategies in (epistemically) indistinguishable states of the system. It is reasonable not to allow this, and to require strategies to be *uniform*. For this, a strategy must correspond to what an agent knows.

Indeed, since [10] various proposals combining epistemic logic and ATL in a way that captures the interplay between agents' actions and their knowledge have been made, such as [12, 16, 11, 2, 4]. We discuss the related work in detail at the end of the paper, but we will point out some basic differences in our work here. The most novel aspect of our logic is that we allow agents with different memory abilities to interact in the same system. Many versions of epistemic ATL, for example [12, 16], consider both full memory and memoryless agents (and implicitly, finite memory agents represented as memoryless agents). But in these logics, every agent in the entire system is assumed to have the same type of memory. It is interesting to consider systems where different agents have different memory abilities. For example, a system could consist of some simple, finite memory agents, interacting with other more sophisticated perfect memory agents. Furthermore, in some settings there may be a group of "friendly" agents with known memory capabilities, and a different group of adversarial agents with unknown memory capabilities. By modelling the friendly agents as limited memory agents and the adversaries as perfect memory agents, we can consider the worst case scenario, and verify security properties of a system.

Besides allowing the combination of agents with different memory capabilities, another novel aspect of our logic is that we allow agents to have arbitrary equivalence relations on histories. Just as agents in epistemic Kripke models are traditionally allowed to have any equivalence relations on the states of a system, our systems allow agents to have any equivalence relation on the histories of the system. For example, we can model an agent who has perfect recall, except she always forgets when the system has been in a specific state. Or we could model an agent who only remembers every other past state. Or an agent could remember everything, until the system enters a certain state, at which point the agent's memory is wiped out. Combined with the fact that different agents are allowed to have different types of equivalence relations on histories, allowing them to have arbitrary equivalence relations as well makes our systems quite general. As far as we know, ours are the only ATL-type systems that allow arbitrary equivalence relations on histories.

The ATL tradition, wherein agents are modelled but not actions, views perfect recall as remembering histories of states, whereas the PDL tradition [9], wherein actions are modelled but not agents, views perfect recall as remembering histories of actions. In our framework, these are available as different memory capabilities. Our definition of perfect recall is therefore different from that in [12, 16, 2], as it does not just consider past states, but also considers the agent's own past actions. We can also model what is elsewhere called perfect recall memory, but we call it 'state-based memory' since it is not as strong as our concept of perfect recall memory. Our definition of perfect recall models agents who remember all the past states of the system, but also remember their own past actions, and can reason about the effects of their actions.

In Section 2, we present epistemic concurrent games structures and we define strategies. In Section 3 we present our alternating-time temporal epistemic logic with uniform strategies. Section 4 compares our work to the tradition in ATL with epistemic operators.

2 Epistemic Concurrent Game Structures

In this section, we present a variation on concurrent game structures. We introduce an indistinguishability relation on the set of states for each agent, which puts a new requirement on the transition relation. This model is appropriate in our setting, because our goal is modelling agents with partial information about the state of the system and the effects of their actions on the outcome. The agents' partial information about the current state is represented by their indistinguishability relations, while the actions they can choose reflect the agents' limited information about the way the system updates.

Definition 1. *An* epistemic concurrent game structure *(ECGS) is a tuple* $\langle Q, \Pi, \Sigma, \mathcal{B}, \sim, \pi, Av, \delta \rangle$ *where*

- Q *is a set of states,*
- Π *is a set of propositions,*
- $\Sigma = \{a_1, ..., a_n\}$ *is a finite set of agents.*
- $\mathcal{B}$ *is a finite set of actions.*
- $\sim$: $\Sigma \to \mathcal{P}(Q \times Q)$ *is an equivalence function associating to each agent* a_i *an equivalence relation* $\sim_i$.
- $\pi : Q \to \Pi$ *is the valuation function,*
- $Av : Q \times \Sigma \to \mathcal{P}(\mathcal{B})$ *is the availability function defining the available actions for an agent in a state, with the requirement that for all* $q_1, q_2 \in Q$ *and all* $a_i \in \Sigma$*:* $Av(q_1, a_i) \neq \emptyset$*, and if* $q_1 \sim_i q_2$ *then* $Av(q_1, a_i) = Av(q_2, a_i)$.
- $\delta : Q \times \Sigma \times \mathcal{B} \to \mathcal{P}(Q)$ *is the transition function, with the* determinacy *requirement that for any* $q \in Q$*, for any* $(b_1, ..., b_n) \in \mathcal{B}^n$ *such that* $b_i \in Av(q, a_i)$ *for* $i = 1, ..., n$*, it is required that* $\bigcap_{i=1}^{n} \delta(q, a_i, b_i)$ *be a singleton.*

The uniformity requirement $Av(q_1, a_i) = Av(q_2, a_i)$ reflects the fact that an agent is aware of what actions are available, so if two states are indistinguishable to the agent, the same actions must be available.

We now define strategies in this setting. The notion of a strategy is dependent on an agent's knowledge about the state of the system: if an agent cannot distinguish two histories, then the agent cannot behave differently in those two histories. Thus, the definition of a strategy for an agent is modular with respect to the agent's equivalence relation on histories. So we will begin by defining strategies, and then we will define three interesting equivalence relations on histories.

In order to define strategies, we must first define histories in ECGS's. We need a few subsidiary definitions first. The following assume an n-agent ECGS $\langle Q, \Pi, \Sigma, \mathcal{B}, \sim, \pi, Av, \delta \rangle$.

Definition 2. *We extend the notion of available actions to a vector of* n *actions. For* $q \in Q$*, let* $Av(q) = \{\langle b_1, b_2, ..., b_n \rangle \in \mathcal{B}^n \mid \forall a_i \in \Sigma, b_i \in Av(q, a_i)\}$.

Definition 3. *For* $q \in Q$ *and* $b^* \in Av(q)$*, define the* b^*-successor of q *as follows:* $Succ(q, b^*) = q'$ *iff* $\bigcap_{i=1}^{n} \delta(q, a_i, b_i) = \{q'\}$.

Definition 4. *In an ECGS* L*, suppose* $h = q_0.b_1^*.q_1.b_2^*.q_2...q_{k-1}.b_k^*.q_k$*, where* $q_j \in Q$ *for* $j \in \{0, ..., k\}$ *and* $b_j^* \in \mathcal{B}^n$ *for* $j \in \{1, ..., k\}$*. Then* h *is a* history for L *if* $q_j = Succ(q_{j-1}, b_j^*)$ *for* $j \in \{1, ..., k\}$*. We denote the set of all histories for* L *as* $Hist(L)$.

Note that all histories are finite, even though infinite executions are possible.

Finally we can define a strategy. For this definition, we extend the Av function from state-agent pairs to history-agent pairs in the obvious way, as the set of actions available at the last state in the history: $Av(q_0.b_1.q_1...q_k, a_i) = Av(q_k, a_i)$.

Definition 5. *Given an ECGS* L*, let* $\approx_i$ *be an arbitrary equivalence relation on* $Hist(L)$*, and* $a_i \in \Sigma$*. A* $\approx_i$ uniform strategy for a_i *is a function* $f_i : Hist(L) \to \mathcal{B}$ *satisfying the following requirements:*

1. *For all* $h \in Hist(L)$*,* $f_i(h) \in Av(h, a_i)$.
2. *If* $h_1 \approx_i h_2$ *then* $f(h_1) = f(h_2)$.

Now that we have given the definition of a strategy with respect to a general equivalence relation, we present several interesting equivalence relations giving rise to different classes of strategies.

Definition 6. *Histories* $h_1 = q_0...q_k$ *and* $h_2 = r_0...r_j$ *are* memoryless equivalent for agent a_i *iff* $q_k \sim_i r_j$*. This is denoted* $h_1 \sim_i^m h_2$*. If* f_i *is a* $\sim_i^m$ *uniform strategy for agent* a_i*, then it is called a* memoryless strategy *for* a_i.

This equivalence relation describes agents who are only aware of the present state but forget everything that has already happened. Next we define strategies for agents who remember the past states of the system.

Definition 7. *For* $h_1, h_2 \in H$ *and* $a_i \in \Sigma$*,* h_1 *and* h_2 *are* state memory equivalent*, written* $h_1 \sim_i^s h_2$ *iff* $h_1 = q_0.q_1...q_k$ *and* $h_2 = r_0.r_1...r_k$ *and for* $j = 0, ..., k$*,* $q_j \sim_i r_j$*. If* f_i *is a* $\sim_i^s$ *uniform strategy for agent* a_i*, then it is called a* strategy with state memory *for* a_i.

This equivalence relation and class of strategies describe agents who remember all the past states of the system, but either do not remember their own actions, or do not reason about the effects of their own actions. Next we give the equivalence relations for agents who remember every state of the system, remember all their own actions, and understand all the effects of their actions.

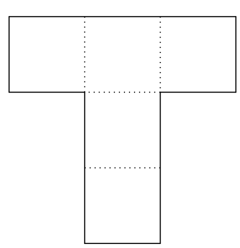
Definition 8. *In an n-agent ECGS, histories h_1 and h_2 are* perfect-recall equivalent *for agent i, written $h_1 \sim_i^{pr} h_2$, iff either $h_1 = q_1$ and $h_2 = q_2$ (where $q_1, q_2 \in Q$) and $q_1 \sim_i q_2$, or $h_1 = q_0.b_1^*.q_1...q_{j-1}.b_j^*.q_j$ and $h_2 = r_0.c_1^*.r_1...r_{j-1}.c_j^*.r_j$ and all of the following conditions hold:*

1. $q_0.b_1^*.q_1...q_{j-1} \sim_i^{pr} r_0.c_1^*.r_1...r_{j-1}$, *and*
2. $q_j \sim_i r_j$, *and*
3. $b_i = c_i$ *where* $b_j^* = \langle b_1, ..., b_n\rangle$ *and* $c_j^* = \langle c_1, ..., c_n\rangle$.

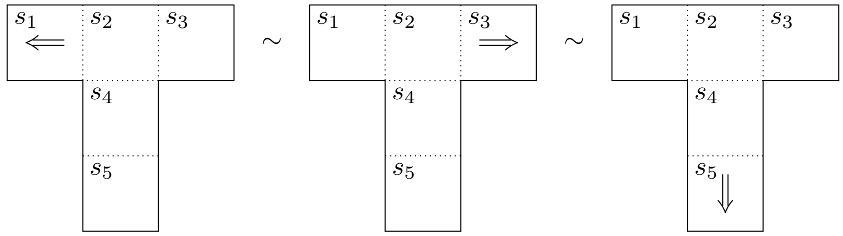
The intuition behind this definition is an agent who remembers its own actions, and can reason about their effects, rather than an agent who just remembers the past states. The perfect recall agent does not observe or remember other agents' actions, however. State memory is often called perfect recall in ATL, whereas our perfect recall is more like PDL perfect recall. In our setting for epistemic ATL there is a real difference between the two. To motivate the definition of perfect recall equivalence, and the differences between the three types of memory we have discussed, consider the following single-agent example.

2.1 Example

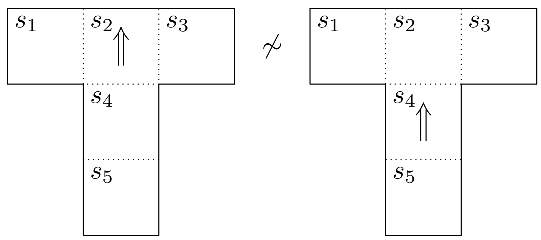
A robot is in a simple maze made up of square spaces. The robot can only perceive whether there are walls immediately in front of, behind and to each side of it, and cannot perceive anything else about the state of the world. The robot has an orientation, either north, south, east or west, and a position in the maze, but the robot is not aware of its orientation and cannot perceive the position, but only the walls around it. We consider the following simple maze:



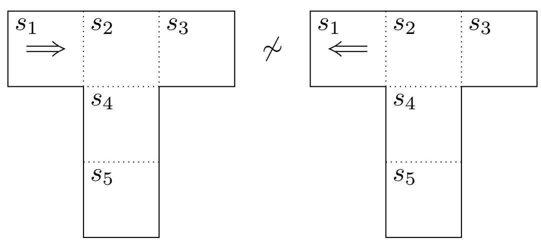
In the following pictures, the arrow represents the robot: both its orientation and its position in the maze. The state of the system consists of the position and orientation of the robot. So, for example, the following three states are indistinguishable:



The following two states, however, are distinguishable, because in the first one, the robot perceives that there is only a wall in front, while in the second one it perceives that there are walls on either side:



Also, the following two states are distinguishable for the robot, because it can distinguish between having an open space in front of it or behind it:



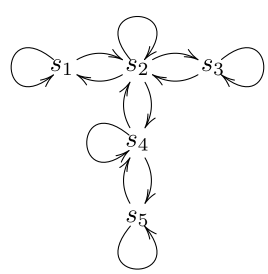
In the following, we denote the states of the system as pairs (s, o) where s is the robot's position in the system (s_1 through s_5), and o is its orientation (n, s, e, or w).

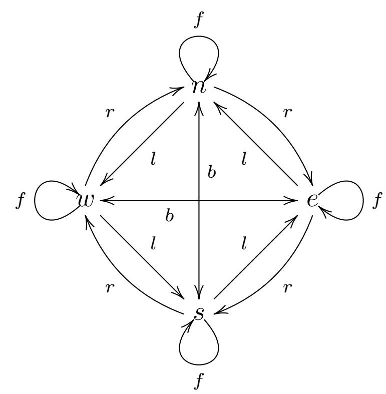
So now we can state the full equivalence relation for the robot:

$$\begin{array}{lll}
(s_1, n) \sim (s_3, s) \sim (s_5, w) & (s_4, n) \sim (s_4, s) & (s_2, n) \\
(s_1, e) \sim (s_3, w) \sim (s_5, n) & (s_4, e) \sim (s_4, w) & (s_2, e) \\
(s_1, s) \sim (s_3, n) \sim (s_5, e) & & (s_2, s) \\
(s_1, w) \sim (s_3, e) \sim (s_5, s) & & (s_2, w)
\end{array}$$

Those in the rightmost column are singleton equivalence classes, since the robot can distinguish the single wall being on its left, right, in front of or behind it.

The robot's actions are go left, go right, go forward or go back, denoted (l, r, f, b). All of these actions are available at every state. The forward action does not change the robot's orientation, but all of the other actions do. Furthermore, the actions change the robot's position if there is space available where the robot tries to go. However, if the robot for example goes left when there is a wall to its left, it changes its orientation but not its position. The following diagram shows the possible transitions between positions and orientations, the combination of which gives the state of the system. The left hand side shows the positions and the right hand side shows the orientations. The arrows in the left hand side are unlabelled because the identity of the transition action between positions depends on the robot's orientation in the starting position: for example, $(s_4, n) \xrightarrow{f} (s_2, n)$ whereas $(s_4, w) \xrightarrow{r} (s_2, n)$.





For clarity, here is part of the transition relation only for position s_1, for any orientation:

	f	r	b	l
(s_1, n)	(s_1, n)	(s_2, e)	(s_1, s)	(s_1, w)
(s_1, e)	(s_2, e)	(s_1, s)	(s_1, w)	(s_1, n)
(s_1, s)	(s_1, s)	(s_1, w)	(s_1, n)	(s_2, e)
(s_1, w)	(s_1, w)	(s_1, n)	(s_2, e)	(s_1, s)

Now, suppose that the robot knows the structure of the maze, but is dropped into a state without knowing its position or orientation. We want to investigate what the robot can achieve by taking actions to explore the system, depending on whether it is a memoryless, state memory, or perfect recall agent.

Suppose the robot starts out in state (s_4, n). Consider the following three sequences of actions.

1. $(s_4, n).b.(s_5, n)$
2. $(s_4, n).f.(s_2, n).l.(s_1, w)$
3. $(s_4, n).f.(s_2, n).r.(s_3, e)$

First of all, suppose the agent is memoryless. Then, the histories 1, 2, and 3 are all equivalent, since the last states are equivalent.

On the other hand, with state memory, the robot can distinguish 1 from 2. In fact, it is easy to see that the robot can distinguish any history that starts in position s_4 and ends in position s_5 from any history that starts in s_4 and ends in s_1. But the agent cannot distinguish 2 from 3. This is because the robot only looks at the past states, and $(s_1, w) \sim (s_3, e)$, as do the first two states in the histories. The robot does not consider its own past actions.

However, if the robot has perfect recall, it can also distinguish 2 and 3, since the two histories have different sequences of actions. Thus, with perfect recall, the agent is allowed to remember its own past actions and distinguish histories based on this information, as well as information about the states.

3 The logic uATEL

In this section we present a logic uATEL for alternating-time temporal epistemic logic with uniform strategies.

Definition 9. *The syntax of* uATEL *is as follows.*

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid K_i\varphi \mid \langle\!\langle A \rangle\!\rangle_\sharp \bigcirc \varphi \mid \langle\!\langle A \rangle\!\rangle_\sharp \Box \varphi \mid \langle\!\langle A \rangle\!\rangle_\sharp \varphi \mathcal{U} \varphi$$

where $p \in \Pi$, $i \in \Sigma$, $A \subseteq \Sigma$, and $\sharp$ is one of a, p and c.

For a single agent a_i, we write $\langle\!\langle a_i \rangle\!\rangle$ rather than $\langle\!\langle \{a_i\} \rangle\!\rangle$. We also use $\wedge$ and $\rightarrow$ in the usual way, defined in terms of $\vee$ and $\neg$. The subindices a, p, c for the coalitional modalities stand for different semantics. Before we give these semantics, we motivate the differences with some further examples.

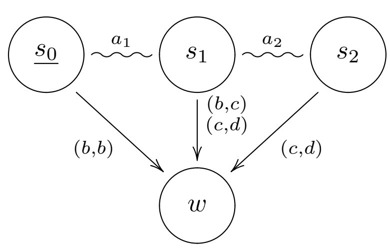
Example 1. There are two agents, a_1 and a_2, and five states: s_0, s_1, s_2, w, and l. $s_0 \sim_1 s_1$ and $s_1 \sim_2 s_2$. There are three actions, b, c, and d, and two propositions, which we also call w (for win) and l (for lose), with $w \models w$ and $l \models l$. The transitions are as follows:

s_0		a_1		
		b	c	d
	b	w	l	l
a_2	c	l	l	l
	d	l	l	l

s_1		a_1		
		b	c	d
	b	l	l	l
a_2	c	w	l	l
	d	l	w	l

s_2		a_1		
		b	c	d
	b	l	l	l
a_2	c	l	l	l
	d	l	w	l

Here is a picture of the system, where all the transitions are available at s_0, s_1, and s_2, and the ones not displayed go to l.



The question is, in state s_0, do the agents have a strategy to reach w? a_2 knows the state is s_0, so she will definitely do b. But a_1 does not know whether the state is s_0 or s_1. If the state were s_1, he would rationally want to do c, since a_2 would now know whether the state was s_1 or s_2, so a_2 would do d to be safe. So in s_0, a_1 wants to do b, but in s_1 a_1 wants to do c. The problem is that a_1 cannot distinguish s_0 and s_1, so he has no strategy to be sure to get the system to reach w. So our semantics must say that

$$s_0 \models \neg\langle\langle\{a_1, a_2\}\rangle\rangle \bigcirc w.$$

Now observe that this reasoning does not hold in the case of the general knowledge semantics, which only considers the union of equivalence relations for the agents in the group, so that in this case from state s_0 we only need to consider that state and s_1, but only with the common knowledge semantics, that it its transitive closure, in this case, all three states s_0, s_1, s_2. This is embodied in the semantics for $\langle\langle\{a_1, a_s\}\rangle\rangle_a$ (where a stands for 'active').

Given are an ECGS L and $q \in L$. For $\Gamma \subseteq \Sigma$, $\sim^*_\Gamma$ is the transitive reflexive closure of $\bigcup_{a_i \in \Gamma} \sim_i$. We use λ to denote an element of Q^+ (where Q is the set of states of L), and $\lambda[i]$ denotes the ith state in the string λ, starting from 0, e.g. if $\lambda = q_0.q_1.q_2$ then $\lambda[1] = q_1$. We also need to define a group strategy:

Definition 10. *For a group of agents $\Gamma \subseteq \Sigma$, we say $F = \{f_a \mid a \in \Gamma\}$ is a* group strategy for Γ *if for all $a_i \in \Gamma$, f_{a_i} is a uniform strategy for a_i.*

Now we can give a possible semantics for the "next" operator.

$(L, q) \models \langle\langle\Gamma\rangle\rangle_a \bigcirc \varphi$ iff there exists a group strategy F_Γ for Γ such that $\forall q' \sim^*_\Gamma q, \forall\lambda \in out(q', F_\Gamma), (L, \lambda[1]) \models \varphi$

This *active coalitional strategy* semantics matches with the intuitive notion of a group of agents having a strategy to reach a goal particularly in settings where all the agents in the group are active in trying to reach the goal, but also have other choices they could make which would prevent the goal from being reached. It is less intuitive in situations where there are agents in the group whose actions cannot affect the outcome of the system from a certain state, as we will see in the following example.

Example 2. Consider a system with two agents a_1 and a_2 and states $\{s_0, ..., s_4\}$, where $s_0 \sim_{a_1} s_1$ but not for a_2, and all the other states can be distinguished by both agents. There is one proposition p, only $s_2 \models p$, and the transitions are $\delta(s_0, a_1, e) = \{s_2, s_3\}$, $\delta(s_0, a_2, b) = \{s_2\}$, $\delta(s_0, a_2, c) = \{s_3\}$, $\delta(s_1, a_1, e) = \{s_4\}$, $\delta(s_1, a_2, d) = \{s_4\}$. So, a_1 has no effect on the execution of the system. At both starting states she can only choose e. Agent a_2, on the other hand, can distinguish s_0 and s_1, and at s_0, he can choose the b action to make p true, but at s_1, a_2 has only one choice and p cannot become true no matter what either agent does. So, we want that $s_0 \models \langle\langle a_2\rangle\rangle \bigcirc p$ but, $s_0 \models \neg\langle\langle\{a_1, a_2\}\rangle\rangle \bigcirc p$.

Let us consider whether it is reasonable that the semantics tells us that $s_0 \models \neg\langle\langle\{a_1, a_2\}\rangle\rangle \bigcirc p$. On the one hand, it is strange to think that while the smaller group consisting only of a_2 can bring about $\bigcirc p$, the larger group $\{a_1, a_2\}$

cannot bring about $\bigcirc p$. In fact, this violates the property in traditional ATL that if $\Gamma_1 \subseteq \Gamma_2$ then $\langle\langle \Gamma_1 \rangle\rangle \bigcirc \varphi \to \langle\langle \Gamma_2 \rangle\rangle \bigcirc \varphi$ [8]. On the other hand, if we think of a group strategy as a strategy where all the agents in the group are active and aware that their actions will reach the outcome, this outcome is less surprising.

In the $\langle\langle \Gamma \rangle\rangle_a$ semantics above we indeed get that $s_0 \models \neg\langle\langle \{a_1, a_2\} \rangle\rangle_a \bigcirc p$. We now propose an alternative $\langle\langle \Gamma \rangle\rangle_p$ semantics (p for 'passive') that achieves $s_0 \models \langle\langle \{a_1, a_2\} \rangle\rangle_p \bigcirc p$.

> $(L, q) \models \langle\langle \Gamma \rangle\rangle_p \bigcirc \varphi$ iff there exists a group strategy F_B for some $B \subseteq \Gamma$ such that $\forall q' \sim_B^* q$, $\forall \lambda \in out(q', F_B)$, $(L, \lambda[1]) \models \varphi$.

This is the *passive coalitional strategy.* In the active strategy operator, the strategy works at all points that any agent considers possible, but in the passive operator, there is a subset of agents who control the strategy, and it works at all states they consider possible, but there are also passive agents in the group whose actions and knowledge make no difference.

We propose a third strategic operator as well. In the active and passive coalition operators the agents can coordinate their actions into a group strategy, but they cannot coordinate or combine their knowledge prior to acting. We would like to also analyze what a group of agents can achieve when they share all the knowledge they possess, as well as acting strategically together. We can model this situation simply by quantifying over the states that are equivalent to the current state for *all* agents in the group (i.e., the accessibility relation for *distributed knowledge* among that group), rather than quantifying over the states that are equivalent for *at least one* agent in the group, and recursively so (i.e., the accessibility relation for *common knowledge* among that group), as in the above semantics. We call this the communication strategy operator and annotate it with a c.

> $(L, q) \models \langle\langle \Gamma \rangle\rangle_c \bigcirc \varphi$ iff there exists a group strategy F_Γ for Γ such that $\forall q' \in \{r \in Q \mid r \sim_i q \; \forall i \in \Gamma\}, \forall \lambda \in out(q', F_\Gamma), (L, \lambda[1]) \models \varphi$.

When the agents share their knowledge, the issue of active and passive strategies no longer arises, because they now coincide.

After these preparations, we now give the complete semantics of uATEL, wherein we have only spelled out the $\Box$ and $\mathcal{U}$ versions for one of the three coalitional modalities (the other two are similar).

Definition 11 (Semantics of uATEL). *Let an ECGS L and a state q in L be given.*

- *for $p \in \Pi$, $L, q \models p$ iff $p \in \pi(q)$,*
- *$L, q \models \neg\varphi$ iff $L, q \not\models \varphi$,*
- *$L, q \models \varphi_1 \vee \varphi_2$ iff $L, q \models \varphi_1$ or $L, q \models \varphi_2$,*
- *$L, q \models K_i\varphi$ iff for all $q' \sim_i q$, $L, q' \models \varphi$.*
- *$L, q \models \langle\langle \Gamma \rangle\rangle_a \bigcirc \varphi$ iff there exists a group strategy F_Γ for Γ such that $\forall q' \sim_\Gamma^* q$, $\forall \lambda \in out(q', F_\Gamma)$, $L, \lambda[1] \models \varphi$,*

– $L, q \models \langle\langle\Gamma\rangle\rangle_a \Box\varphi$ *iff there exists a group strategy* F_Γ *for* Γ *such that* $\forall q' \sim^*_\Gamma q$, $\forall\lambda \in out(q', F_\Gamma)$, $L, \lambda[n] \models \varphi$ *for all* $n \geq 0$,
– $L, q \models \langle\langle\Gamma\rangle\rangle_a \varphi_1 \mathcal{U} \varphi_2$ *iff there exists a group strategy* F_Γ *for* Γ *such that* $\forall q' \sim^*_\Gamma q$, $\forall\lambda \in out(q', F_\Gamma)$, *there exists* $m \in \mathbb{N}$ *such that* $L, \lambda[m] \models \varphi_2$ *and for all* $0 \leq n \leq m$, $L, \lambda[n] \models \varphi_1$,
– $L, q \models \langle\langle\Gamma\rangle\rangle_p \bigcirc \varphi$ *iff there exists a group strategy* F_B *for some* $B \subseteq \Gamma$ *such that* $\forall q' \sim^*_B q$, $\forall\lambda \in out(q', F_B)$, $L, \lambda[1] \models \varphi$,
– $L, q \models \langle\langle\Gamma\rangle\rangle_c \bigcirc \varphi$ *iff there exists a group strategy* F_Γ *for* Γ *such that* $\forall q' \in \{r \in Q \mid r \sim_i q \text{ for all } i \in \Gamma\}$, $\forall\lambda \in out(q', F_\Gamma)$, $L, \lambda[1] \models \varphi$,

If φ *holds at all states in all EGCS's, then we write* $\models \varphi$ *(for '*φ *is valid').*

Some elementary results for this semantics are as follows.

Proposition 1. *For all sets of agents* Γ *and for all formulas* φ*:* $\models \langle\langle\Gamma\rangle\rangle_a\varphi \rightarrow \langle\langle\Gamma\rangle\rangle_p\varphi$ *and* $\models \langle\langle\Gamma\rangle\rangle_p\varphi \rightarrow \langle\langle\Gamma\rangle\rangle_c\varphi$.

Proposition 2. *If* $\Gamma_1 \subseteq \Gamma_2$, $\models \langle\langle\Gamma_1\rangle\rangle_p\varphi \rightarrow \langle\langle\Gamma_2\rangle\rangle_p\varphi$ *and* $\models \langle\langle\Gamma_1\rangle\rangle_c\varphi \rightarrow \langle\langle\Gamma_2\rangle\rangle_c\varphi$.

Whereas this is *false*: "$\models \langle\langle\Gamma_1\rangle\rangle_a\varphi \rightarrow \langle\langle\Gamma_2\rangle\rangle_a\varphi$ implies $\models \langle\langle\Gamma_1\rangle\rangle_c\varphi \rightarrow \langle\langle\Gamma_2\rangle\rangle_c\varphi$." An obvious embedding is the following. We use $\models_{ATL}$ for the ATL semantics, and $\langle\langle\Gamma\rangle\rangle$ as the ATL coalitional operator. A *perfect information system* M is an ECGS such that for all agents i, for all states q and q', $q \sim_i q'$ iff $q = q'$.

Proposition 3. $M, q \models_{ATL} \langle\langle\Gamma\rangle\rangle\varphi$ *if and only if* $M, q \models \langle\langle\Gamma\rangle\rangle_a\varphi$.

3.1 Example

In this section, we present an extended example based on the following scenario. Consider a game played by two agents using a deck of cards with all the face cards (J,Q,K,A) removed. The deck is shuffled and each agent is given one card. Each agent sees their own card without revealing it to the other agent. Then each agent has the choice of trading their card for a different one from the deck, once, or keeping their card. The agents' goal is for the sum of their cards to be at least seven.

First we model this game as an ECGS. We model it as a three agent system, where a_1 and a_2 represent the two agents playing the game and the third agent, *env* represents the environment, resolving choices that would otherwise be nondeterministic. We define the set of states as

$$Q = \{(x, y, z) \mid x \in \{i, f\} \text{ and } y, z \in \{2, 3, ..., 10\}\},$$

where (i, y, z) represents an initial state where a_1 has card y and a_2 has card z, and (f, y, z) represents a final state, after the agents have decided whether to swap their cards, where a_1 has card y and a_2 has card z. The equivalence relations are as follows:

$$\begin{aligned}
(x_1, y_1, z_1) \sim_1 (x_2, y_2, z_2) &\text{ iff } x_1 = x_2 \text{ and } y_1 = y_2\\
(x_1, y_1, z_1) \sim_2 (x_2, y_2, z_2) &\text{ iff } x_1 = x_2 \text{ and } z_1 = z_2\\
(x_1, y_1, z_1) \sim_{env} (x_2, y_2, z_2) &\text{ iff } x_1 = x_2 \text{ and } y_1 = y_2 \text{ and } z_1 = z_2
\end{aligned}$$

The set of actions is $\mathcal{B} = \{swap, stay\} \cup \{(x, y) \mid x, y \in \{2, ..., 10\}\}$. The transition relation is as follows:

$$\begin{aligned}
\delta((i, y, z), a_1, stay) &= \{(f, y, z') \mid z' \in \{2, ..., 10\}\} \\
\delta((i, y, z), a_1, swap) &= \{(f, y', z') \mid y', z' \in \{2, ..., 10\}\} \\
\delta((i, y, z), a_2, stay) &= \{(f, y', z) \mid y' \in \{2, ..., 10\}\} \\
\delta((i, y, z), a_2, swap) &= \{(f, y', z') \mid y', z' \in \{2, ..., 10\}\} \\
\delta((i, y, z), env, (y', z')) &= \{(f, y, z), (f, y, z'), (f, y', z), (f, y', z')\} \\
\delta((f, y, z), a, stay) &= \{(f, y, z)\} \text{ for all } a \in \Sigma
\end{aligned}$$

Thus, in an initial state, each agent chooses whether to keep their card or change it. If they change their card, the environment picks a new card for them. Then, in a final state, all three agents only have one action available, *stay*, which does not change the state (we only include this because our semantics require infinite runs).

Finally, our only proposition will be w, representing that the agents win, and $\pi((x, y, z)) = \{w\}$ iff $x = f$ and $y + z \geq 7$. Otherwise $\pi((x, y, z)) = \emptyset$.

Now we will investigate which formulas are true at certain states in this system. First, consider the state $(i, 8, 1)$: the first agent has an 8 and the second agent has 1. Intuitively, would we say that the group consisting of both agents has a winning strategy from this state? Of course,

$$(i, 8, 1) \models \neg\langle\!\langle a_2 \rangle\!\rangle_a \bigcirc w,$$

because acting alone a_2 has no strategy to ensure that they reach a winning state, but

$$(i, 8, 1) \models \langle\!\langle a_1 \rangle\!\rangle_a \bigcirc w,$$

because a_1 can use the strategy of keeping his card and be sure to win. But notice that

$$(i, 8, 1) \models \neg\langle\!\langle \{a_1, a_2\} \rangle\!\rangle_a \bigcirc w,$$

since, for example $(i, 1, 1) \sim_2 (i, 8, 1)$ and there is no strategy from $(i, 1, 1)$ for $\{a_1, a_2\}$ to achieve w in the next state. However, it is true that

$$(i, 8, 1) \models \langle\!\langle \{a_1, a_2\} \rangle\!\rangle_p \bigcirc w,$$

because $\{a_1\} \subseteq \{a_1, a_2\}$, and no matter what a_2 does, a_1 has a winning strategy. So there is a *passive* strategy for a_1 and a_2 to reach w at the next state, because a_2 has an active strategy for this goal, and nothing a_1 does can interfere with this accomplishment, so a_1 passively brings about w at the next state. And of course,

$$(i, 8, 1) \models \langle\!\langle \{a_1, a_2\} \rangle\!\rangle_c \bigcirc w.$$

Intuitively, this is because the agents share their knowledge and then decide on a strategy, so they both know that keeping their cards is a good strategy.

To highlight the differences between $\langle\langle\Gamma\rangle\rangle_c$ and the other two strategic operators, consider the state $(i, 4, 5)$. Here, we have both

$$(i, 4, 5) \models \neg\langle\langle\{a_1, a_2\}\rangle\rangle_a \bigcirc w, \text{ and } (i, 4, 5) \models \neg\langle\langle\{a_1, a_2\}\rangle\rangle_p \bigcirc w.$$

Intuitively, this means that the agents do not have enough information about their current state to have either an active or a passive group strategy to reach a winning state. However, if the agents share all their information, they realize that both of them keeping their cards is a good group strategy. Thus,

$$(i, 4, 5) \models \langle\langle\{a_1, a_2\}\rangle\rangle_c \bigcirc w.$$

In terms of the semantics, this is because there are no other states that are equivalent to $(i, 4, 5)$ for both a_1 and a_2, so the group strategy only needs to guarantee the desired outcome at this single state.

4 Related Work

Uniform strategies In this paper we have only considered uniform strategies, since we are considering what agents are able to accomplish, and an agent must choose their actions based on their own knowledge. Non-uniform strategies, however, may be useful sometimes, for example for analyzing worst-case scenarios where agents could perhaps secretly communicate or otherwise gain unexpected knowledge. Only [10], the first paper about ATEL, does not consider uniform strategies. Interestingly, in the original paper on ATL there is a discussion of ATL with incomplete information, and uniform strategies are defined [3, p.706–710]. However, in their approach the agents' equivalence relations are defined in terms of propositions (i.e., valuations) rather than in terms of states, leading to many restrictions on the expressible formulas and making the logic quite complicated. It is well-known that in multi-agent Kripke models such an identification of states with valuations is very restrictive for the expressivity of a logic.

De re or de dicto A second major aspect of ATEL-type logics is whether they allow *de dicto* or *de re* strategies. A *de re* strategy to achieve something is a uniform strategy that will succeed starting from any state the agent considers possible. A *de dicto* strategy to achieve something, on the other hand, is a uniform strategy that will succeed from the present state, but not from every state the agent considers possible. So if an agent has a *de re* strategy to achieve something, the agent knows that he has the strategy and knows what the strategy is. But if the agent has a *de dicto* strategy, he does not know what the strategy is. Note that a *de dicto* strategy is in general uniform- even though it does not succeed from all the states the agent considers possible, it requires the agent to take the same action in all states that are equivalent for him.

In the current paper, we only consider *de re* strategies, as we are concerned with what agents can be sure to achieve based on their knowledge. While *de dicto* strategies are interesting from an outside perspective, they are not useful to the agents inside the system, trying to achieve certain goals. Like our logic, ATOL

[12] and ATL with perfect and imperfect information and recall [16] can only express *de re* abilities, whereas other logics can only express *de dicto* abilities, for example Epistemic Coalition Logic in [2]. Some logics can express both *de dicto* and *de re* abilities. For example, in Constructive Strategic Logic [11], the basic group operator expresses *de dicto* ability, but combining this operator with a special epistemic operator expresses *de re* ability. In [4], six variants of epistemic ATL are considered, with both *de dicto* and *de re* abilities. It is shown that the expressiveness of ATL with *de re* abilities ("subjective abilities") and ATL with *de dicto* abilities ("objective abilities") are incomparable, both in the perfect recall and imperfect recall cases. Similar results are shown for ATL* with *de re* and *de dicto* abilities.

Coalitional operators In logics with uniform strategies the semantics for the coalitional operator (followed by next) has the following generic form

> $L, q \models \langle\langle \Gamma \rangle\rangle \bigcirc \varphi$ iff there exists a group strategy F_Γ such that $\forall q' \sim^? q$, $\forall \lambda \in out(q', F_\Gamma),\ L, \lambda[1] \models \varphi$,

where the definition of $\sim^?$ is variable. Most often, the relation is either $\bigcup_{a \in H} \sim_a$ (general knowledge for group of agents H), as in [16], or it is $(\bigcup_{a \in H} \sim_a)^*$, the transitive closure of $\bigcup_{a \in H} \sim_a$ (common knowledge for H). We have seen in Example 1 that the former is not felicitous. Similarly, in [4], six varieties of ATL and six varieties of ATL* are compared, and the semantics of the ability operators in all the varieties of the logics are defined using the union of the equivalence relations of the agents (i.e., general knowledge). It would be interesting to know if defining the semantics in terms of the common knowledge relation would change the results presented in that paper.

In ATOL [12] the semantics of the ability operator is much more subtle. The operator is defined as follows, where A and Γ are groups of agents and $\mathcal{K}$ is either C, E or D, for common knowledge, general knowledge and distributed knowledge, respectively.

> $L, q \models \langle\langle A \rangle\rangle_{\mathcal{K}(\Gamma)} \bigcirc \varphi$ iff there is a group strategy F_A such that $\forall q' \sim^{\mathcal{K}}_{\Gamma} q, \forall \lambda \in out(q', S_A),\ L, \lambda[1] \models \varphi$.

The ability operator in this logic is very powerful: not only does it subsume both the union relation semantics (which can be expressed as $\langle\langle A \rangle\rangle_{E(A)}$) and the common knowledge semantics (which can be expressed as $\langle\langle A \rangle\rangle_{C(A)}$), it is even possible to define the ability of one group of agents with respect to the knowledge of another group of agents. We have the following correspondence with our logic.

Proposition 4. *Consider a ECGS L with memoryless agents, and $q \in L$.*

1. $L, q \models \langle\langle \Gamma \rangle\rangle_a \varphi$ *if and only if* $L, q \models_{ATOL} \langle\langle \Gamma \rangle\rangle_{C(\Gamma)} \varphi$.
2. $L, q \models \langle\langle \Gamma \rangle\rangle_p \varphi$ *if and only if* $\exists \Gamma' \subseteq \Gamma$ *such that* $L, q \models_{ATOL} \langle\langle \Gamma' \rangle\rangle_{C(\Gamma')} \varphi$.
3. $L, q \models \langle\langle \Gamma \rangle\rangle_c \varphi$ *if and only if* $L, q \models_{ATOL} \langle\langle \Gamma \rangle\rangle_{D(\Gamma)} \varphi$.

In item 2, since we consider a finite set of agents, it follows that $L, q \models \langle\langle \Gamma \rangle\rangle_p \varphi$ if and only if $L, q \models_{ATOL} \bigvee_{\Gamma' \subseteq \Gamma} \langle\langle \Gamma' \rangle\rangle_{C(\Gamma')} \varphi$. So, for memoryless systems, our logic can be translated into ATOL. However, our system can deal with non-memoryless systems as well. Even for memoryless systems ATOL can express properties that our logic cannot express, such as $\langle\langle \Gamma_1 \rangle\rangle_{E(\Gamma_2)} \bigcirc \varphi$, where $\Gamma_2 \not\subseteq \Gamma_1$.

Memory Abilities Another difference among the various logics is whether they allow perfect recall or not. Traditional ATL [3] allows agents to have perfect recall, although it was shown in [4] that in the perfect information setting, for ATL it does not matter whether agents have perfect recall or not — this only matters in the case of ATL*. In [10] only perfect recall agents are considered. In [16], four different classes of operators are considered: *IR* for perfect information and recall, *iR* for imperfect information and perfect recall, *Ir* for perfect information and imperfect recall, and *ir* for imperfect information and recall. These different levels of abilities determine which strategies are considered admissible. Interestingly, in the logic of [16] it is possible to combine different ability operators within the same formula, for example $\langle\langle A \rangle\rangle_{iR} \bigcirc \langle\langle B \rangle\rangle_{Ir} \Box \varphi$ means that group A has an imperfect information perfect recall strategy so that at the next state group B will have a perfect information imperfect recall strategy to make φ always true. While being able to express such formulas is interesting, it is not clear what the meaning of them is—for example, if some of the agents are in both groups A and B in the above formula, it means that sometimes they are being considered as memoryless agents and sometimes as perfect recall agents. The logic ATOL [12] is mostly concerned with memoryless agents.

Combining memory abilities One of the new aspects of our work is the ability to represent models with agents of different ability in the same system and in the same logic. We do this by treating an agent's memory abilities as part of the underlying system rather than as an aspect of the semantics of the logic. This is similar to the way that each agent's knowledge is traditionally encoded in the system as an arbitrary equivalence relation on states, but now we encode an agent's knowledge as an equivalence relation on histories rather than on states. So, rather than being an aspect of the logic, the agent's memory ability becomes an aspect of the system. This makes it possible to discuss agents with different memory abilities in the same formula, which is impossible in the other varieties of epistemic ATL. For example, we can have a formula such as $\langle\langle ab \rangle\rangle \Box \varphi$ where a is an agent with perfect recall and b is a memoryless agent. This formula is not expressible in other logics.

5 Conclusion and Future Work

We have presented a logic for reasoning about the abilities of agents to cooperate to achieve a goal when they are uncertain about the state of the world. Our systems allow different agents to have different memory abilities. We presented a new definition of perfect recall, which takes the history of states and the history of actions into account.

We intend further to study the properties of this logic, such as decidability and complexity. For example, in [5], it is proven that model checking is undecidable for a variant of epistemic ATL with strategies based on common knowledge.

We also wish to be able to describe memory abilities in the logical language. We are further contemplating dynamic operators for change of memory ability, and other levels of cooperation than the three considered in this paper.

Yet another future direction is that the logic uATEL may help to pave the way to a coalitional event logic. Pauly's game logic [14] corresponds to the next temporal fragment of ATEL, and this game logic is subsumed by the coalitional announcement logic (CAL) of [1]. Coalitional announcements are *public* events enacted by coalitions.

Acknowledgements
We thank the reviewers for their comments. Hans van Ditmarsch is also affiliated to the IMSc, Chennai, as associated researcher. This research is supported by European Research Council grant EPS 313360.

References

1. Ågotnes, T., van Ditmarsch, H.: Coalitions and announcements. In: Proc. of 7th AAMAS. pp. 673–680. IFAAMAS (2008)
2. Ågotnes, T., Alechina, N.: Epistemic coalition logic: completeness and complexity. In: AAMAS. pp. 1099–1106 (2012)
3. Alur, R., Henzinger, T.A., Kupferman, O.: Alternating-time temporal logic. J. ACM 49(5), 672–713 (2002)
4. Bulling, N., Jamroga, W.: Comparing variants of strategic ability: how uncertainty and memory influence general properties of games. Autonomous Agents and Multi-Agent Systems 28(3), 474–518 (2014)
5. Diaconu, R., Dima, C.: Model-checking alternating-time temporal logic with strategies based on common knowledge is undecidable. Applied Artificial Intelligence 26(4), 331–348 (2012)
6. Emerson, E.A., Clarke, E.M.: Using branching-time temporal logic to synthesize synchronization skeletons. Sci. Comput. Programm. 2, 241–266 (1982)
7. Emerson, E.A., Halpern, J.Y.: "sometimes" and "not never" revisited: on branching versus linear time temporal logic. J. ACM 33(1), 151–178 (1986)
8. Goranko, V., van Drimmelen, G.: Complete axiomatization and decidability of alternating-time temporal logic. Theor. Comp. Science 353(13), 93 – 117 (2006)
9. Harel, D., Kozen, D., Tiuryn, J.: Dynamic Logic. MIT Press, Cambridge MA (2000), foundations of Computing Series
10. van der Hoek, W., Wooldridge, M.: Cooperation, knowledge, and time: Alternating-time temporal epistemic logic and its applications. Stud. Log. 75(1), 125–157 (2003)
11. Jamroga, W., Ågotnes, T.: Constructive knowledge: what agents can achieve under imperfect information. Journal of Appl. Non-Classical Logics 17(4), 423–475 (2007)
12. Jamroga, W., van der Hoek, W.: Agents that know how to play. Fundam. Inform. 63(2-3), 185–219 (2004)
13. Kripke, S.: Semantical analysis of modal logic. Zeitschrift fur Mathematische Logik und Grundlagen der Mathematik (1963)
14. Pauly, M.: A modal logic for coalitional power in games. Journal of Logic and Computation 12(1), 149–166 (2002)
15. Pnueli, A.: The temporal logic of programs. In: The Eighteenth Annual IEEE Symposium on Foundations of Computer Science. pp. 46–57 (1977)
16. Schobbens, P.Y.: Alternating-time logic with imperfect recall. Electr. Notes Theor. Comput. Sci. 85(2), 82–93 (2004)