THE REVIEW OF SYMBOLIC LOGIC
Volume 4, Number 3, September 2011

# REASONING WITH PROTOCOLS UNDER IMPERFECT INFORMATION

ERIC PACUIT
Tilburg Institute for Logic and Philosophy of Science (TiLPS)
Department of Philosophy, Tilburg University

SUNIL SIMON
Centrum voor Wiskunde & Informatica (CWI)

**Abstract.** We introduce and study a PDL-style logic for reasoning about protocols, or plans, under imperfect information. Our paper touches on a number of issues surrounding the relationship between an agent's abilities, available choices, and information in an interactive situation. The main question we address is under what circumstances can the agent commit to a protocol or plan, and what can she achieve by doing so?

**§1. Introduction and motivation.** There is a growing literature using different (multi-) modal logics to reason about communities of agents engaged in some form of social interaction. In particular, various combinations of temporal logics, epistemic and doxastic logics, action logics, and preference logics have been studied in this context.[1] A key issue that has emerged is how best to represent and reason about the underlying *protocol* that governs the agents' interactions in a particular social situation.

Intuitively, a *protocol* describes what the agents "can" or "cannot" do (say, observe) in a social interactive situation. This leads to *substantive* assumptions[2] about the formal model, such as which actions (observations, messages, utterances) are available (permitted) at any given moment. These assumptions can be roughly categorized according to the different uses of "can":

1. To describe physical, temporal, or historical possibilities: A typical example is the assumption an agent *cannot* receive a message unless another agent sent it earlier. Such assumptions limit the options available to the agents at any given moment.
2. To describe the agents' abilities, or skills: The options available to an agent at any given moment are defined not only by what is "physically possible," but also by the agent's *capacity* to perform various actions. For example, "Ann *can* throw a bull's-eye" typically means that Ann has the ability to (repeatedly) throw a bull's-eye.
3. To describe compliance to some type of norm: The social or conversational[3] norms at play in the interactive situation being modeled (i.e., the "rules of the game") impose

Received: September 22, 2010.

[1] A complete survey of these "logics of rational agency" is outside the scope of this paper. The interested reader can consult (van Benthem, 2010; van der Hoek & Wooldridge, 2003b; Meyer & Veltman, 2007) for a discussion and the relevant references.

[2] See Roy & Pacuit (2011) for a general discussion of "substantive assumptions" in the context of epistemic models of games.

[3] See Parikh & Ramanujam (2003, sec. 6) for a discussion of Gricean norms in this context.

© Association for Symbolic Logic, 2011

doi:10.1017/S1755020311000190

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

further constraints on the options available to each agent. For example, common conversational rules include: "Do not blurt everything out at the beginning"; "Do not repeat yourself"; "Let others speak in turn"; and "Be honest." Imposing such rules *restricts* the legitimate sequences of possible statements.

So, a protocol encodes not only which options are *feasible*, but also what is *permissible* for the agents to do or say. Of course, an interesting and important component of a logical analysis of rational agents is to disambiguate these different meanings of "can" (cf. Horty, 2001; van der Hoek *et al*., 1998; Elgesem, 1997; Governatori & Rotolo, 2005; Cross, 1986). In this paper, we take a more abstract perspective in which a protocol simply identifies a subtree from the "grand stage" of all possible sequences of events that could take place in an interactive situation.

A number of authors have forcefully argued that the underlying protocol is an important component of any analysis of (social) interactive situations and should be explicitly represented in a formal model (cf. Fagin *et al*., 1995; van Benthem *et al*., 2009; Parikh & Ramanujam, 2003; Hoshi, 2009; van Eijk *et al*., 2009; van Otterloo, 2005). Indeed, much of the work over the past 20 years using epistemic logic to reason about distributed algorithms has provided interesting case studies highlighting the interplay between protocol analysis and epistemic reasoning (an important example here is the seminal paper by Halpern & Moses, 1990, on the "generals problem").

The central question of this paper is what do the agents "know" about the underlying protocol, and how is this reflected in the logic used to reason about social interactions? A typical assumption is that there is a fixed global protocol that all the agents have (explicitly or implicitly) agreed to follow (and this is commonly known). This is the assumption in the *epistemic temporal logics*, as discussed by Parikh & Ramanujam (2003), Halpern & Moses (1990), van Benthem *et al*. (2009, sec. 4), among many others (Fagin *et al*., 1995; van Benthem, 2010, are textbook presentations of this literature). These logical systems use linear or branching time models with added epistemic structure induced by the agents' different capacities for observing events. The models provide a "grand stage" where histories of some social interaction unfold constrained by an underlying *protocol*. Thus, the protocol is represented *extensionally* in the models as a set of histories (sequences of events).[4] From the point of view of the logical systems that have been developed to reason about these structures (e.g., as discussed in Halpern *et al*., 2004; van Benthem & Pacuit, 2006; van Benthem *et al*., 2009), the protocol is only implicitly represented, for example, with statements of the form "$F\phi$" meaning that "$\phi$ is true at some moment in the future (after the agents perform actions consistent with the protocol)."

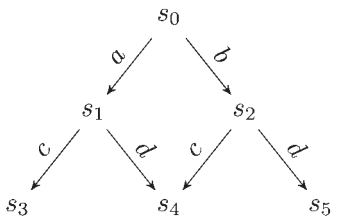
In this paper, we develop a logical framework where protocol(s) are "first-class citizens" (cf. van Benthem, 2001a). This provides a local perspective where simple protocols can be combined to construct more complex ones. Thus, we drop the assumption that there is a single fixed protocol and consider situations where the protocol is created "as needed." A number of authors have suggested different variations of *propositional dynamic logic* (PDL) to reason about protocols, or *strategies*, from this local, "constructive" point of view (e.g., see, Fagin *et al*., 1995; van Benthem, 2001a; van Benthem & Pacuit, 2006; van Eijk *et al*., 2009; Wang, 2010). The idea is that PDL action expressions explicitly describe different protocols. Under this interpretation, the PDL formula $[\pi]\phi$ has the interpretation

[4] Cf. van der Meyden (1996), where the models are generated by unfolding some multiagent finite state machine.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

"$\phi$ is guaranteed to be true by following the protocol $\pi$." Here, "following the protocol $\pi$" means that agent(s) makes choices so that the resulting sequence of events matches[5] $\pi$.

We start with a single agent who, in each possible state, can choose from a finite set of actions (the actions she "can" perform in the sense of points 1 and 2 above). The many agent case is discussed in §5. Each action corresponds to a (possibly nondeterministic) transition from the current state to a new state, and there may be different actions available at different states. In other words, we assume that the agent is in a *labeled transition system*, which we call an **arena**. The arena describes the actions that are available at each state and the possible consequences of each action. The following is an example of an arena:




A **protocol** is a tree with labels from the (finite) set of possible actions. We are interested in what properties the agent(s) can *guarantee* will be true by *adopting a given protocol*. The idea is that adopting a protocol at a state restricts the paths that the agent will follow from that state. In general, adopting a protocol does not commit the agent(s) to a single course of action, but, rather, focuses the agent's(s') attention on the "relevant" decision problems. Thus, "adopting a protocol" simply amounts to "committing to a *plan*,"[6] something that is crucial for an autonomous (rational) agent (this is argued most forcefully by Bratman, 1987). In his influential book, Michael Bratman argues, *inter alia*, that

> plans help make deliberation tractable for limited beings like us. They provide a clear, concrete purpose for deliberation, rather than merely a general injunction to do the best. They narrow the scope of the deliberation to a limited set of options. And they help answer a question that tends to remain unasked within traditional decision theory, namely; where do decision problems come from?(Bratman, 1987, p. 33)

One contribution of our paper is to explore the conditions under which agent(s) can engage in such (future-directed) planning (cf. Cohen & Levesque, 1990; Meyer *et al.*, 1999). We focus on *structural properties* of the interactive situation (i.e., what the agents *can* do) and what the agents "know" about the decision problems they face. We leave for future work how to incorporate the agents' *motivating attitudes* (e.g., desires, goals, wishes) into our logical analysis. Thus, we focus on when the agent(s) *can* (implicitly or explicitly) agree to adopt a protocol, or commit to a plan, instead of why the agent(s) would *want* to agree to a protocol, or plan.

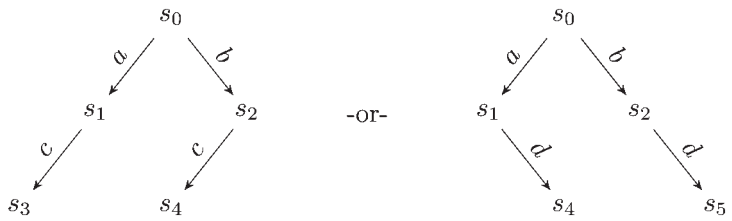
Our first observation is that it is important to interpret the PDL actions expressions over *finite trees* rather than *paths*. In other words, our basic actions expressions denote finite trees instead of the usual one-step actions (cf. Ramanujam & Simon, 2009). For example, suppose that the agent is in state $s_0$ in the above arena and consider the protocol "either

[5] Here we are thinking of $\pi$ as a *regular expression*: See Harel *et al.* (2000) for a discussion.

[6] In what follows, we will use "protocol" and "plan" interchangeably.

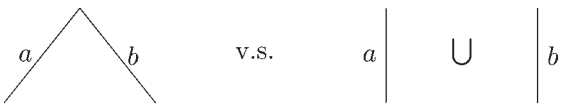
https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

choose $c$ or choose $d$." This protocol gives only partial information about what actions to follow at a given state (e.g., the protocol does not offer any advice about what to do at $s_0$). This protocol can be described by the PDL expression $(a \cup b); c \cup (a \cup b); d$. Note that every path in the above arena is consistent with this protocol, so we can say that this protocol is *enabled* at $s_0$. However, as van Benthem (2010) points out, this way of thinking about the protocol misses a crucial point: The agent must commit to do either $c$ or $d$ *independent* of which action is chosen at state $s_0$. In other words, by committing to this protocol (at $s_0$), the agent must choose between the following two restrictions on future choices:




This distinction is not important if we are interested in only the states that can result by following this protocol—in this case, $\{s_3, s_4\} \cup \{s_4, s_5\}$. However, it is becomes important when constructing complex plans from simpler ones using the regular operations of PDL (union $\cup$, concatenation ;, and Kleene star $^*$) or if an agent conditions on the plans of another agent (or her future self).

An interesting feature of allowing branching in atomic programs is that we can represent a choice between $a$ and $b$ in two different ways. The picture on the left denotes the atomic tree consisting of two branches, one labeled with $a$ and the other with $b$. The picture on the right is a complex program built using the union operator from two atomic trees, each containing only one branch.

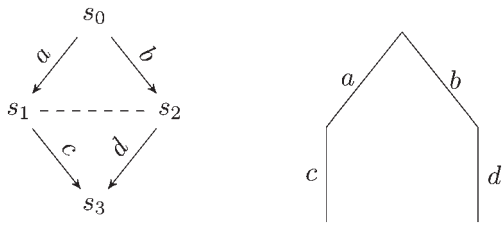



These two programs have very different interpretations corresponding to different ways of understanding what it means for an agent to commit the plan: do $a$ or do $b$. On the first interpretation, the agent commits to choosing between actions $a$ or $b$ *when the time comes* (possibly ignoring the other options that may be available to the agent at that moment). On the second interpretation, the agent must choose between two future courses of actions: doing $a$ or doing $b$. The point is that $a$ and $b$ each may lead to a different set of states.

Our main contribution in this paper is to analyze different ways in which a protocol "can" be adopted (by either a single agent or a group of agents) taking each agent's *point of view* into account. Since we assume that actions may be nondeterministic, there may be many ways in which a protocol can be "realized" at a position in an arena. This creates uncertainty for the agent since, in general, she may not know which state results from a particular action. However, there may be other sources of imperfect information. For example, the agent may have only limited memory or observational power, or the agent may be uncertain about the exact "starting position" or initial state of the situation. Thus, at certain positions in the arena, for whatever reason, it may appear to the agent that she is in a different position or set of positions. For example, consider the following situation

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

where the agent cannot distinguish between nodes $s_1$ and $s_2$ and the protocol pictured to the right (do $a$ followed by $c$ or do $b$ followed by $d$):




This protocol is clearly enabled in the situation without the uncertainty relation between $s_1$ and $s_2$. However, in the above situation at $s_0$, the agent cannot agree to "*knowingly*" follow the protocol since she is uncertain about the actions that are available at states[7] $s_1$ and $s_2$.

Sections 2 and 3 introduce our formal models (an *arena* (with imperfect information) and a *protocol*) and discusses two key definitions: what it means for a protocol to be *enabled* (Definition 2.4) and what it means to be *subjectively enabled* (Definition 3.3). Section 4 develops a PDL-style logic for reasoning about what agents can achieve in arenas by committing to protocols, or plans, with a complete axiomatization provided in Theorem 4.13. Section 6.2 compares our logic to similar logical frameworks, and Section 5 focuses on extending our analysis to the many-agent situation.

**§2. Preliminaries: arenas and protocols.** The definitions in this section are standard and are included to make the paper self contained and to fix notation (the key notions are Definitions 2.2, 2.3, and 2.4).

**Basic protocols**. As discussed in the previous section, *protocols* are finite labeled trees. We first settle on notation for finite trees. Let $\Sigma$ be a finite set whose elements are called **actions**. A $\Sigma$-labeled (finite) tree $T$ is a tuple $(S, \{\Rightarrow_a\}_{a\in\Sigma}, s_0)$ where $S$ is a (finite) set of nodes, $s_0 \in S$ is the root, and for each $a \in \Sigma$, $\Rightarrow_a \subseteq S \times S$ is the edge relation satisfying the usual properties (we write $s_i \Rightarrow_a s_j$ for $(s_i, s_j) \in \Rightarrow_a$):

1. *irreflexive*: for each $a \in \Sigma$ and $s \in S$, it is not the case that $s \Rightarrow_a s$;
2. *antisymmetric*: for each $a \in \Sigma$ and $s, t \in S$, if $s \Rightarrow_a t$ then it is not the case that $t \Rightarrow_a s$; and
3. *unique predecessor*: for each $s \in S$ with $s \neq s_0$ there is a unique $t$ such that $t \Rightarrow_a s$ for some $a$.

For a node $s \in S$, let $\mathcal{A}(s) = \{a \in \Sigma \mid \exists s' \in S \text{ where } s \Rightarrow_a s'\}$ denote the set of *actions available at* $s$. A node $s$ is called a *leaf node* if $\mathcal{A}(s) = \emptyset$, and the set of all leaf nodes in the tree is denoted by $frontier(T)$. For a set $X$ and a finite sequence $\rho = x_1 x_2 \dots x_m \in X^*$, let $last(\rho) = x_m$ denote the last element in this sequence and $first(\rho) = x_1$ the first element. We extend this notion to a set $Y \subseteq X^*$ as $last(Y) = \{x \mid \exists \rho \in Y \text{ with } last(\rho) = x\}$. The following definition is standard:

DEFINITION 2.1 (Paths). *A* **path** *in the tree* $T = (S, \{\Rightarrow_a\}_{a\in\Sigma}, s_0)$ *is an alternating sequence of nodes and actions* $\rho = s_0 a_0 s_1 a_1 \cdots a_{k-1} s_k$ *satisfying the following condition:*

[7] Alternatively, we can say that the agent forgets at state $s_1$ (and $s_2$) the choice that was made at state $s_0$.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

*for all* $j : 0 \leq j < k$, *we have* $s_j \Rightarrow_{a_j} s_{j+1}$. *The* **length** *of a path* $\rho$, *denoted* $len(\rho)$, *is the number of actions appearing in* $\rho$. *A path* $\rho$ *is* **maximal** *in* $T$ *if* $first(\rho) = s_0$ *and* $\mathcal{A}(last(\rho)) = \emptyset$. *Let* $Paths(T)$ *denote the set of all maximal paths in* $T$. *For* $\rho = s_0 a_0 s_1 a_1 \ldots s_k$, *let* $head(\rho) = a_0$ *and* $tail(\rho) = s_1 a_1 \ldots s_k$.

In some cases, it is convenient to define a path as a sequence of states (or actions). For example, we say a sequence of *states* $\sigma = s_0 s_1 \cdots s_k$ is a **path of states** if there are actions $a_0, \ldots, a_{k-1}$ such that $s_0 a_0 s_1 a_1 \cdots a_{k-1} s_k$ is a path (define a **path of actions** similarly). We can use these definitions to define the **height** of a finite tree $T$ (the length of the longest path): $height(T) = \max\{len(\rho) \mid \rho \in Paths(T)\}$. Note that the above labeled trees may be *nondeterministic* since two edges from the same node can have the same label (i.e., there may be distinct nodes $s, s'$, and $s''$ such that $s \Rightarrow_a s'$ and $s \Rightarrow_a s''$). However, if the tree is intended to represent a *protocol* or *plan* that an agent has committed to follow, then it is natural to restrict attention to *deterministic* trees:

DEFINITION 2.2 (Basic Protocol). *A finite tree* $T = (S, \{\Rightarrow_a\}_{a\in\Sigma}, s_0)$ *is called a* **(basic) protocol** *if it is deterministic: for each* $s, s', s'' \in S$ *and* $a \in \Sigma$, *if* $s \Rightarrow_a s'$ *and* $s \Rightarrow_a s''$ *then* $s' = s''$.

**Finite arenas**. We model an interactive (or decision-theoretic) situation in a standard way as a labeled transition system, which we call an **arena** (or *finite state machines*, following standard terminology in theoretical computer science literature, or a *frame*, following standard terminology in the modal logic literature).

DEFINITION 2.3 (Finite Arena). *Let* $W$ *be a nonempty finite set, whose elements are called* **positions** *or* **states**, *and* $\Sigma$ *a finite set of basic actions. An* **arena** *is a structure* $\mathcal{G} = (W, \{\rightarrow_a\}_{a\in\Sigma})$ *where for each* $a \in \Sigma$, $\rightarrow_a \subseteq W \times W$. *Following standard notation, we write* $w \rightarrow_a v$ *if* $(w, v) \in \rightarrow_a$.

The above notation for available actions and paths (Definition 2.1) is readily applied to finite arenas. Finite arenas are "third-person" models of the interactive situation describing:

1. *all* available choices for the agent(s) at each state (for each state $s$, this is the set $\mathcal{A}(s)$); and
2. the sequence of *all possible* decision problems the agent(s) will encounter (via the transitions given by $\rightarrow_a$ for each $a \in \Sigma$).

A protocol or plan *restricts* the available choices for the agent(s). Intuitively, if an agent agrees to follow a finite protocol, then she commits to restricting her choices to all and only those actions compatible with the protocol. Of course, not all protocols *can* be followed in any situation. This leads us to the key notion of a protocol being **enabled** at a state $u$ in an arena. If there is no uncertainty in the arena, then the formal definition of a protocol being enabled is completely straightforward: A protocol $T$ is enabled at $u$ in $\mathcal{G}$ if $T$ can be embedded in the unwinding of $\mathcal{G}$ at $u$. We give the formal details of this definition below.

- Suppose that $T = (S, \{\Rightarrow_a\}_{a\in\Sigma}, s)$ and $T' = (S', \{\Rightarrow'_a\}_{a\in\Sigma}, s')$. We say that $T$ can be **embedded** in $T'$, denoted $T \sqsubseteq T'$, if there is an injective function $f : S \rightarrow S'$ such that for all $a \in \Sigma$ and $s, t \in S$, $s \Rightarrow_a t$ iff $f(s) \Rightarrow'_a f(t)$.
- Suppose that $T = (S, \{\Rightarrow_a\}_{a\in\Sigma}, s)$ can be embedded in $T' = (S', \{\Rightarrow'_a\}_{a\in\Sigma}, s')$ (with embedding $f$). The tree[8] $(f[S], \{\Rightarrow''_a\}_{a\in\Sigma}, f(s))$ where for $a \in \Sigma$, $\Rightarrow''_a$ is the

[8] Recall that for $X \subseteq S$, $f[X] = \{f(s) \mid s \in X\}$.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

relation $\Rightarrow'_a \cap (f[S] \times f[S])$ is called a **restriction** of $T'$ to $T$ and is denoted $T' \restriction T$. If $T$ is a protocol and $T'$ and arbitrary tree, then, since $T$ is deterministic and $T'$ is nondeterministic, there may be more than one embedding of $T$ into $T'$. In such a case, let the *union* of the restrictions be *the* restriction of $T'$ by $T$.

- Let $\mathcal{G} = (W, \{\rightarrow_a\}_{a\in\Sigma})$ be an arena. The **unwinding**, or **tree unfolding**, of $\mathcal{G}$ at state $u$ the tree $T_u = (S_u, \{\Rightarrow^u_a\}_{a\in\Sigma}, s_u)$ where 1. $S_u$ is the set of all paths of nodes starting at $u$ ($S_u = \{x_0x_1\cdots x_n \mid$ for each $i = 0, \ldots, n$ $x_i \in W$ where $x_0 = u$ and $x_0x_1\cdots x_n$ is a path of states in $\mathcal{G}\}$ (note that Definition 2.1 can be applied to arenas as well as to trees), 2. $ux_1\cdots x_n \Rightarrow^u_a ux_1\cdots x_nx_{n+1}$ iff $x_n \rightarrow_a x_{n+1}$, and 3. $s_u = (u)$ (i.e., the path consisting of the single state $u$). Note that, in general, the tree unfolding $T_u$ will be a nondeterministic tree.

DEFINITION 2.4 (Enabled). *Suppose that $T$ is a basic protocol and that $\mathcal{G} = (W, \{\rightarrow_a\}_{a\in\Sigma})$ is an arena. We say that $T$ is* **enabled at** *$u$, denoted $enabled(t, u)$, if $T$ can be embedded in $T_u$.*

Intuitively, if a protocol $T$ is enabled at a state $u$ in an arena $\mathcal{G}$, then it is (physically, objectively) *possible* for the agent to *agree* to follow $T$. Of course, this does not necessarily mean that the agent *knows* (or *believes*) that she can follow $T$; the agent *wants* to follow $T$; or that it is in the agent's interest to follow $T$.

**§3. Imperfect information.** A protocol being enabled simply means that the protocol is *feasible*—that is, physically possible. In this section, we explore a different sense in which a protocol is "possible," one that takes into account the agent's *point of view*. Our first task is to extend the definition of an arena with an explicit representation of the agent's "point of view" at each position in the arena. As is standard in the epistemic logic literature, we use a relation on the set of states in an arena to represent the agent's uncertainty about her position in the arena. In general, there are many *sources* of this uncertainty: For example,

1. if action $a$ is nondeterministic, then the agent may be uncertain about which state will result by choosing $a$;
2. the agent may have some prior (partial) information about the interactive situation; or
3. the agent may be limited in what she can observe and what she remembers.

In many situations, it is interesting to distinguish between these different sources, but for now, we simply describe the agent's point of view at each state.

DEFINITION 3.1 (Arena with Imperfect Information). *An* **arena with imperfect information** *is a structure $\mathcal{G}^I = (W, \{\rightarrow_a\}_{a\in\Sigma}, \rightsquigarrow)$ where $(W, \{\rightarrow_a\}_{a\in\Sigma})$ is a finite arena and $\rightsquigarrow \subseteq W \times W$.*

For each position $u$, let $\mathcal{I}(u) = \{w \mid u \rightsquigarrow w\}$ be the agent's "*point of view*." A useful way of thinking about the $\rightsquigarrow$ relation is as special "$\epsilon$-transitions"[9] (well studied in the automata-theoretic literature). They represent transitions that the agent does not have control over, and so they cannot be ruled out by committing to a protocol or plan. An

[9] We are very grateful to R. Ramanujam, who suggested (among other things) this way of thinking about the agent's "uncertainty" in the context of our paper.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

important conceptual point is that $\rightsquigarrow$ is not the same "type" of transition as the $\Sigma$-labeled transitions: Rather than the agent deciding whether to follow such a transition, $\epsilon$-transitions are externally imposed "silent" transitions that *generate* uncertainty.

The above models do not impose any structural properties on the action and $\rightsquigarrow$ relations. However, a number of properties discussed in the literature are relevant. Suppose that the agent is in position $w$ but "thinks" she is in position $v$ (i.e., $w \rightsquigarrow v$), and consider an action $a \in \mathcal{A}(w) \cap \mathcal{A}(v)$. In this case, the agent is aware that she can do $a$ and will not fail. Furthermore, unless there is a "miracle," doing action $a$ should not remove the agent's "uncertainty" (e.g., the $\rightsquigarrow$ relation). Formally,

- **No Miracles**: For all $a \in \Sigma$ and all $w, v, w', v' \in W$, if $w \rightsquigarrow v$, $w \rightarrow_a w'$, and $v \rightarrow_a v'$, then $w' \rightsquigarrow v'$.

Imposing no miracles means that the basic actions are assumed to be "uninformative." No miracles covers the situation when $a \in \mathcal{A}(w) \cap \mathcal{A}(v)$ (recall that $\mathcal{A}(w)$ is the set of actions available at $w$). The remaining interesting situations are when an action $a$ is available only in one of the states. First, if $a \in \mathcal{A}(w)$, but $a \notin \mathcal{A}(v)$, then the agent does not realize that $a$ is actually available. Second, if $a \in \mathcal{A}(v)$, but $a \notin \mathcal{A}(w)$, then the agent believes that she can do $a$, but will fail[10] if she attempts to execute this action. Formally, these situations are:

- **Success**: If $w \rightsquigarrow v$, then $\mathcal{A}(v) \subseteq \mathcal{A}(w)$.
- **Awareness**: If $w \rightsquigarrow v$, then $\mathcal{A}(w) \subseteq \mathcal{A}(v)$.

Of course, if $\rightsquigarrow$ is symmetric, then these properties are equivalent and we have $\mathcal{A}(w) = \mathcal{A}(v)$ provided $w \rightsquigarrow v$. These properties address the relationship between the actions available at the current state (which the agent may not have access to) and the actions available at states the agent considers "possible" (via $\rightsquigarrow$). The next property focuses on the relationship between the actions available at the set of states the agent considers "possible." If $w \rightsquigarrow v$ and $w \rightsquigarrow v'$, then the agent may find herself in either $v$ or $v'$ and so should face the same decision problem:

- **Certainty of available actions**: If $w \rightsquigarrow v$ and $w \rightsquigarrow v'$, then $\mathcal{A}(v) = \mathcal{A}(v')$.

Of course, these properties are all equivalent in the important special case when the agent's $\epsilon$-transition is an equivalence relation (a common assumption in the epistemic logic and game theory[11] literature). When $\rightsquigarrow$ is an equivalence relation, we follow standard notation and write $\sim$ for $\rightsquigarrow$. This special case is particularly interesting since it helps position our work within the broad literature using various combinations of modal logics to reason about game/decision-theoretic situations (cf. Lorini *et al*., 2009; van Benthem, 2001b). We will return to these properties throughout the paper but do not commit ourselves to any of them at this point.

[10] Note that we do not address in this paper what happens (from the agent's point of view) if she tries to do an action $a$ that is not actually available (i.e., the agent *attempts* action $a$). This interesting situation will be addressed in future work. See Lorini & Herzig (2008) for a very interesting discussion relevant to this situation.

[11] Of course, game theorists tend to focus on arenas that are themselves *trees*—that is, extensive games with imperfect information.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

For a protocol $T$ and a position $u$, the notion of $T$ being enabled (Definition 2.4) at $u$ is well defined for an arena with imperfect information. However, as discussed above, this is an *objective* notion from the modeler's point of view that does not take into account that the agents may be imperfectly informed about their "location" in the arena. What we need is a *subjective* version of Definition 2.4. One idea is to mimic the restriction operation of Definition 2.4, but to ensure *at each step* that we take into account all and only the positions that the agent has access to via the $\rightsquigarrow$ relation. Intuitively, a protocol $T$ is *subjectively enabled* at a position $u$ in an arena with imperfect information if:

1. the agent is *certain that* $T$ is enabled (for all $v \in \mathcal{I}(u)$, $T$ is enabled at $v$); and
2. the agent will be certain that she is, in fact, following the protocol at *every stage* of the protocol.

This second point is important as there is a difference between "knowing that a protocol is enabled" and "being able to *knowingly follow* a protocol."[12] This difference is crucial for an agent contemplating committing to a long-term plan.[13] Thus, our definition must take into account the forest $\{T_v \mid v \in \mathcal{I}(u)\}$ *for every position u not ruled out by the protocol*.

According to Definition 2.4, $enabled(T, u)$ is true if there is an embedding of $T$ into $T_u$. We have to complicate this simple picture in the presence of imperfect information. We start by stating the most general definition and then show how to simplify it in the presence of the structural assumptions discussed above (e.g., assuming $\rightsquigarrow$ is an equivalence relation[14]). First of all, note that in arenas with imperfect information, the restriction of a protocol $T$ is not a tree, but, rather, a *forest* (possibly containing trees of different heights). Thus, we need to introduce notation for forests in an arena. Let $\mathcal{G}$ be an arena (with imperfect information). First, recall that the notion of a path (Definition 2.1) applies to arenas and, by assumption, the last element of a path is always a state. We say that a path $\rho$ is an *initial segment* of $\rho'$ if $\rho'$ is $\rho$ followed by a possibly empty path. Formally, $\rho = w_0 a_0 \cdots a_{k-1} w_k$ is an initial segment of $\rho'$ if there is an $i \geq 0$ such that $\rho' = w_0 a_0 \cdots a_{k-1} w_k a_{k+1} \cdots a_{k+i-1} w_{k+i}$. Given a set of paths $X$ that is closed under initial segment, we define an edge relation in the obvious way: $\rho \Rightarrow_a^X \rho'$ iff $\rho = w_0 a_0 \cdots a_{k-1} w_k$ and $\rho' = w_0 a_0 \cdots a_{k-1} w_k a w$. A set of paths $X$ from an arena $\mathcal{G}$ that is closed under initial segment is called a **forest in** $\mathcal{G}$ if $\{\Rightarrow_a^X\}_{a \in \Sigma}$ satisfies the properties 1, 2, and 3 in the definition of a tree given above.

It is not hard to see that if a protocol $T$ is enabled at $u$, then the restriction of $T$ at $u$ gives us a forest $X$ with each path in $X$ is associated with a node in $T$. Generalizing to situations with imperfect information, we may need to associate more than one path with a node in $T$. Thus, we define the restriction of $T$ in an arena with imperfect information to be a forest $X$ and function mapping paths in $X$ onto nodes in $T$:

DEFINITION 3.2 (Subjective Restriction). *Let* $\mathcal{G}^I = (W, \{\rightarrow_a\}_{a \in \Sigma}, \rightsquigarrow)$ *be an arena with imperfect information,* $u \in W$ *and* $T = (S, \{\Rightarrow_a\}_{a \in \Sigma}, s_0)$ *a protocol. The* **subjective**

[12] See Broersen (2008) for a discussion related to this point.

[13] After all, an agent cannot commit to a temporally extended plan if she is certain now that she will not be able to choose in a way that is consistent with that plan. Of course, this does not preclude the possibility that the agent may need to revise or drop her plan *even after committing to it* (perhaps because she learned that the plan is no longer feasible) (Icard *et al*., 2010).

[14] The reader interested only in this special case can use the statement of Lemma 6.1 in place of the definition given below.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

**restriction** *of $T$ in $(\mathcal{G}^I, u)$, denoted $(\mathcal{G}^I, u) \restriction_s T$, is a pair $(X, f)$ where $X$ is a forest in $\mathcal{G}^I$ and $f$ is a function from $X$ onto $S$. Both $X$ and $f$ are defined inductively as follows:*

0. *$X_0 = \mathcal{I}(u)$ ($v \in X_0$ is understood as a one-element sequence) and for all $v \in X_0$, set $f_0(v) = s_0$*
n. *Suppose $X_n$ and $f_n$ have been constructed, for each $\rho \in S_n$, for all $a \in \mathcal{A}(f_n(\rho))$, let $Y_a = \{\rho a w \mid last(\rho) \to_a w \text{ in } \mathcal{G}^I\} \cup \bigcup\{\mathcal{I}(w) \mid last(\rho) \to_a w \text{ in} \mathcal{G}^I\}$. Define*

$$X_{n+1} = X_n \cup \bigcup_{a \in \mathcal{A}(f_n(\rho)), \rho \in X_n} Y_a$$

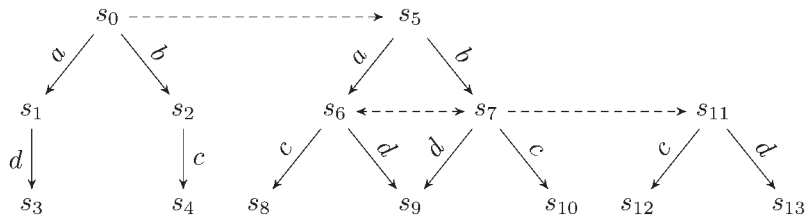
*Let $f_{n+1}$ extend $f_n$ such that for each new node $\rho a w \in Y_a$, set $f_{n+1}(\rho a w) = s'$ where $f_n(\rho) \Rightarrow_a s'$ in $T$.*

*Let $X = X_{height(T)}$ and $f = f_{height(T)}$. Finally, define the* **frontier of** *$(\mathcal{G}^I, u) \restriction_s T$ as follows: $frontier((\mathcal{G}^I, u) \restriction_s T) = \{last(\rho) \in W \mid \mathcal{A}(f(\rho)) = \emptyset\}$.*

Note that since $T$ is deterministic, $f_{n+1}$ is well defined. Define the actions available at a path in a forest as follows: suppose that $X$ is a forest and $\rho \in X$ and define $\mathcal{A}(\rho) = \{a \in \Sigma \mid \text{there is a } \rho' \in X \text{ such that } \rho \Rightarrow_a^X \rho'\}$.

DEFINITION 3.3 (Subjectively Enabled). *A protocol $T$ is* **subjectively enabled** *at $u$ in $\mathcal{G}^I = (W, \to, \rightsquigarrow)$, denoted s-enabled$(T, (\mathcal{G}^I, u))$, if the structure $(\mathcal{G}^I, u) \restriction_s T = (X, f)$ satisfies the condition $\forall \rho \in X, \mathcal{A}(\rho) = \mathcal{A}(f(\rho))$.*

Notice that without additional structural assumptions on $\rightsquigarrow$, a protocol being subjectively enabled does *not* imply that the protocol is enabled. For example, consider the arena below and the protocol discussed in the Introduction: "either do $a$ followed by $c$ or do $b$ followed by $d$." This protocol is subjectively enabled but not enabled at state $s_0$.




(*Note that the protocol is still subjectively enabled if we impose the no miracle property, which would add a number of $\rightsquigarrow$ edges, represented by dashed arrows.*)

We conclude this section with two observations. The first is that in situations of *perfect information*, subjectively enabled is equivalent to enabled:

PROPOSITION 3.4. *Suppose that $\mathcal{G}^I = (W, \{\to_a\}_{a \in \Sigma}, \rightsquigarrow)$ satisfies the property that for all $w \in W$, $\mathcal{I}(w) = \{w\}$. Then, for any protocol $T$ and state $w \in W$, $T$ is enabled at $w$ in $(W, \{\to_a\}_{a \in \Sigma})$ iff $T$ is subjectively enabled at $w$ in $\mathcal{G}^I$.*

The proof follows by unpacking the definitions and is left to the reader. Additional structural properties can further simplify the definition of subjectively enabled. We have already remarked that a protocol being "subjectively enabled" at a state $w$ is, in general, not equivalent to the agent knowing that the protocol is enabled at $w$ (i.e., the protocol is objectively enabled according to Definition 2.4 at every state in $\mathcal{I}(w)$) A simple argument shows that these notions coincide when the agent is certain of her available actions and the actions are not informative:

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

LEMMA 3.5. *Suppose that* $\mathcal{G}^I = (W, \{\rightarrow_a\}_{a\in\Sigma}, \rightsquigarrow)$ *satisfies certainty of actions and no miracles. Then, the agent knows that* $t$ *is enabled at* $u$ *iff* $t$ *is subjectively enabled at* $u$.

**§4. What can be achieved with protocols?** An arena with imperfect information describes what *can happen* in an interactive situation both objectively (from the modeler's point of view) and subjectively (from the agent's point of view via the $\rightsquigarrow$ relations). That is, they describe both what is physically possible for the agent to do and what she thinks she can do in an interactive situation. We have not yet addressed what the agent is *able* to do in an interactive situation. In this section, we focus on a different sense of "can" that takes into account the agent's "abilities." We study a number of logical systems that describe what can be *achieved* in an interactive situation.

What can be achieved in an interactive situations depends on the protocol or plan that the agent is currently following. Thus far, we have focused only on *basic protocols*. It is convenient to give an explicit syntax for describing basic protocols.

DEFINITION 4.6 (Syntax for Protocols). *Let* $\mathcal{V}$ *be a countable set of node variables. A* **protocol expression** *is inductively defined as follows:*

- *For each* $x \in \mathcal{V}$, $(x)$ *is a protocol expression.*
- *Suppose that* $J = \{a_1, \ldots, a_m\}$ *is a set of (distinct) actions and for each* $a_i$ *we have a (unique) protocol expression* $t_{a_i}$. *Then,*

$$(x, a_1, t_{a_1}) + \cdots + (x, a_m, t_{a_m})$$

*is a protocol expression where* $x$ *is a new variable not appearing in* $t_{a_i}$.

*Let* $\mathcal{P}(\mathcal{V})$ *denote the set of protocol expressions.*

The idea is that the expression $(x, a, t_a)$ denotes the subtree where $x$ is the root and there is an $a$-edge from $x$ to the subtree described by $t_a$. Note that this syntax generates only deterministic trees (i.e., basic protocols) since each action $a$ in a protocol expression is associated with only one subtree. Of course, there are other ways to syntactically describe finite trees, but the particular choice of syntax is not crucial for our analysis. The important point is that each syntactic expression $t \in \mathcal{P}(\mathcal{V})$ corresponds to an finite tree $T_t$:

DEFINITION 4.7 (Interpretation of Protocol Expressions). *Given* $t \in \mathcal{P}(\mathcal{V})$, *we can inductively define the* **basic protocol** $T_t$ **generated by** $t$ *as follows:*

- *if* $t = (x)$, *then let* $T_t = (S_t, \Rightarrow_t, s_x)$ *where* $S_t = \{s_x\}$ *and* $\Rightarrow_t = \emptyset$.
- *if* $t = (x, a_1, t_{a_1}) + \cdots + (x, a_k, t_{a_k})$, *then inductively we have trees* $T_1, \ldots T_k$ *where for* $j : 1 \leq j \leq k$, $T_j = (S_j, \Rightarrow_j, s_j)$. *Define* $T_t = (S_t, \Rightarrow_t, s_x)$ *where* $s_x$ *is a new state and*

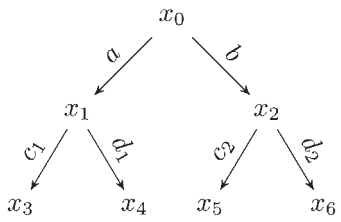
  *(i)* $S_t = \{s_x\} \cup S_{T_1} \cup \ldots \cup S_{T_k}$.
  *(ii)* $\Rightarrow_t = (\bigcup_{j=1,\ldots,k} \Rightarrow_j) \cup \{s_x \Rightarrow_{a_j} {}_j s_j \mid 1 \leq j \leq k\}$.

For $t \in \mathcal{P}(\mathcal{V})$, we often abuse notation and identify $t$ with $T_t$. The following example illustrates the above construction:

The next step is a syntax for describing *complex protocols*. To keep things simple, we focus on the regular operations familiar from action logics such as PDL: Let $\Sigma$ be a finite set of basic actions, and define $\Gamma$ to be the smallest set of expressions generated by the following grammar:

$$t \mid \pi_1; \pi_2 \mid \pi_1 \cup \pi_2 \mid \pi^*$$

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press




The syntactic representation of this tree using Definition 4.6 is:

- $t = (x_0, a, t_1) + (x_0, b, t_2)$ where
  - $t_1 = (x_1, c_1, (x_3)) + (x_1, d_1, (x_4))$ and
  - $t_2 = (x_2, c_2, (x_5)) + (x_2, d_2, (x_6))$.

where $t \in \mathcal{P}(\mathcal{V})$ is a basic protocol (using actions from $\Sigma$). Note that we do not include tests in our language. Adding tests raises a number of interesting issues (many have been extensively discussed in the literature on knowledge programs, see (Fagin *et al.*, 1995; Fagin *et al.*, 1997); however, we leave this extension for future work.[15] We can easily adopt the standard interpretation of these operations to our setting:

1. $\pi_1; \pi_2$ is the protocol where the agent first adopts the protocol $\pi_1$ and then (no matter what happens) adopts the protocol $\pi_2$;
2. $\pi_1 \cup \pi_2$ is the protocol where the agent must first choose which of the two protocols to adopt; and
3. $\pi^*$ is the protocol where the agent continues with protocol $\pi$ any finite number of times (including zero).

Of course, there may be other natural operations in this context, such as "merging"[16] or "revising" (cf. Icard *et al.*, 2010).

Committing to a basic protocol $T$ *restricts* the choices available to the agent, but there is a trade-off: It also *increases* the agent's ability of the agent to *guarantee* that certain propositions are true. Formally, each basic protocol is associated with a set of states $X$ (the *frontier* of $T$ in an arena). The agent can "force" the situation to end up in these states by making choices consistent with the protocol. There are a number of ways to make precise what it means for an agent to "guarantee" that some proposition is true because she adopts the protocols $T$. One option is to see what is true no matter what the agent does, as long as it is consistent with $T$. A second option recognizes that $T$ still represents choices for the agent that will be settled in the course of the interaction. In this case, we are interested in what the agent can force by doing something consistent with $T$. The situation is even more interesting when the agent commits to a complex protocol. If the protocol involves the operators $\cup$ or Kleene star, then the agent first must choose which set of states she wants to have the ability to force. For example, consider the protocol $T_1 \cup T_2$; in order to commit to this protocol, the agent must choose which of the two basic protocols to follow. More generally, given a complex protocol $\pi$, the agent must first decide both *how* to go about adopting $\pi$ and then make her choices "in the moment" consistent with this plan.

This discussion suggests that our basic modality will be interpreted as a sequence of *two* quantifiers (each corresponding to the different "types" of decisions the agent makes when committing to a protocol). This is familiar from other modal logics of ability

[15] Note that there is nothing inherently difficult about adding tests to our language; and, indeed, the results in this paper can be adapted to this situation. We do not include them here to simplify the setting and focus on issues that are orthogonal to issues that are relevant when tests are in the language.

[16] There is an extensive discussion of this using PDL in Wang (2010).

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

(cf. Balbiani *et al*., 2007) and game logics (Parikh, 1985). Of the four possible combinations of quantifiers, we take the following two as primitive (corresponding to $\exists\forall$ and $\exists\exists$, respectively):

- $\langle\pi\rangle^{\forall}\alpha$: By adopting the protocol $\pi$, $\alpha$ is guaranteed to be true.
- $\langle\pi\rangle^{\exists}\alpha$: By adopting the protocol $\pi$, the agent can do something consistent with the protocol that will make $\alpha$ true.

As usual, the remaining two possible combinations of quantifiers are dual to these. We take "adopting a protocol" to mean that the agent decides how to follow the protocol (so an existential quantifier over the different sets of states the agent can force). The second quantifier is over the different ways that the agent actually implements the protocol. These notions are objective since they do not take into account the fact that the agent may be imperfectly informed about her current position in the arena. This suggests the following "epistemized" versions of the above operators:

- $\langle\pi\rangle^{\Box}\alpha$: By *agreeing* to adopt the protocol $\pi$, the agent is certain that $\alpha$ is guaranteed to be true.
- $\langle\pi\rangle^{\Diamond}\alpha$: By *agreeing* to adopt the protocol $\pi$, the agent is can "knowingly" do something consistent with the protocol that will make $\alpha$ true.

***4.1. Epistemic protocol logic.*** Let $\mathsf{At}$ be a countable set of atomic propositions and $\Gamma$ a set of protocol expressions as defined in Definition 4.7 (based on basic actions $\Sigma$). The **epistemic protocol language** is the smallest set $\mathcal{L}_{EPL}$ of formulas generated by:

$$p \in \mathsf{At} \mid \neg\alpha \mid \alpha_1 \vee \alpha_2 \mid \Box\alpha \mid \langle\pi\rangle^{\exists}\alpha \mid \langle\pi\rangle^{\forall}\alpha \mid \langle\pi\rangle^{\Box}\alpha \mid \langle\pi\rangle^{\Diamond}\alpha$$

where $\pi \in \Gamma$. By convention, let $\top = p \vee \neg p$, $\Diamond\alpha = \neg\Box\neg\alpha$, $[\pi]^{\exists}\alpha = \neg\langle\pi\rangle^{\forall}\neg\alpha$, $[\pi]^{\forall}\alpha = \neg\langle\pi\rangle^{\exists}\neg\alpha$, $[\pi]^{\Diamond}\alpha = \neg\langle\pi\rangle^{\Box}\neg\alpha$, and $[\pi]^{\Box}\alpha = \neg\langle\pi\rangle^{\Diamond}\neg\alpha$. We discussed the four protocol modalities above. The remaining modality $\Box$ quantifies over states accessible (in one step) via the $\rightsquigarrow$ relation. Thus, it describes what is true from the agent's point of view. As usual, models are arenas with valuation functions:

DEFINITION 4.8 (Model). *Let* $\mathcal{G}^I = (W, \{\rightarrow_a\}_{a\in\Sigma}, \rightsquigarrow)$ *be an arena with imperfect information (cf. Definition 3.1). A* **model** *based on* $\mathcal{G}^I$ *is a structure* $(W, \{\rightarrow_a\}_{a\in\Sigma}, \rightsquigarrow, V)$ *where and* $V : \mathsf{At} \to 2^W$ *a valuation function.*

Before defining truth in a model, we must "interpret" complex protocols. The idea is to associate with each protocol $\pi$ the collection of states that the agent can force by following $\pi$. Formally, we define sets $R_\pi^{\mathcal{Q}} \subseteq W \times 2^W$ for $\mathcal{Q} \in \{\exists, \forall, \Box, \Diamond\}$ by induction on the structure of $\pi$. We start with the atomic protocols.

**Atomic Protocols**. For an atomic protocol expressions $t$, and $\mathcal{Q} \in \{\exists, \forall, \Box, \Diamond\}$, we define the relation $R_t^{\mathcal{Q}} \subseteq W \times 2^W$ as follows:

- $R_t^{\mathfrak{F}} = \{(u, X) \mid \mathit{enabled}(T_t, u)$ and $\mathit{last}(\mathit{frontier}(T_u \restriction T_t)) = X\}$ (for $\mathfrak{F} \in \{\exists, \forall\}$).
- $R_t^{\Box} = \{(u, X) \mid \mathit{s\text{-}enabled}(T_t, u)$ and $\mathit{last}(\mathit{frontier}((\mathcal{G}, u) \restriction_s T_t)) = X\}$.

The definition of $R_t^{\Diamond}$ is more complicated. The issue is that, in this case, the way the agent implements the protocol must take into account the agent's imperfect information.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

This suggests the following notion: given a path $\rho = s_t^0 a_0 s_t^1 \dots s_t^k \in \mathit{Paths}(t)$, the **subjective path** defined by $\rho$ on the structure $(\mathcal{G}, u) \upharpoonright_s t = (S, \Rightarrow, f)$ is the sequence $\mathfrak{S}(\rho, u) = Z_0 Z_1 \dots Z_k$ where for all $j : 0 \leq j \leq k$, $Z_j = \{s \in S \mid f(s) = s_t^j\}$. We now have

- $R_t^{\Diamond} = \{(u, X) \mid \mathit{s\text{-}enabled}(T_t, u)$ and $\exists \rho \in \mathit{Paths}(T_t)$ with $\mathfrak{S}(\rho, u) = Z_0 Z_1 \dots Z_k$ and $X = Z_k\}$.

**Composition**.

- $R^{\exists}_{\pi_1;\pi_2} = \{(u, X) \mid \exists Y \subseteq W$ such that $(u, Y) \in R^{\exists}_{\pi_1}$ and $\exists v_j \in Y$ such that $(v_j, X) \in R^{\exists}_{\pi_2}\}$.
- for $\mathfrak{F} \in \{\forall, \Box, \Diamond\}$,
  - $R^{\mathfrak{F}}_{\pi_1;\pi_2} = \{(u, X) \mid \exists Y = \{v_1, \dots, v_k\}$ such that $(u, Y) \in R^{\mathfrak{F}}_{\pi_1}$ and $\forall v_j \in Y$, there exists $X_j \subseteq X$ such that $(v_j, X_j) \in R^{\mathfrak{F}}_{\pi_2}$ and $\bigcup_{j=1,\dots,k} X_j = X\}$.

Note that in the definition above, we can assume the set $Y$ is finite since our models are finitely branching. The definition of union and Kleene star is standard (though some care must be taken in the latter case to use a fixed point definition):

**Union**. For $\mathcal{Q} \in \{\exists, \forall, \Box, \Diamond\}$, $R^{\mathcal{Q}}_{\pi_1 \cup \pi_2} = R^{\mathcal{Q}}_{\pi_1} \cup R^{\mathcal{Q}}_{\pi_2}$.

**Iteration**.

- $R^{\exists}_{\pi^*} = \bigcup_{n \geq 0} (R^{\exists}_{\pi})^n$.

For $\mathcal{Q} \in \{\forall, \Box, \Diamond\}$, it is tempting to define iteration as $R^{\mathcal{Q}}_{\pi^*} = \bigcup_{n \geq 0} (R^{\mathcal{Q}}_{\pi})^n$. However, this definition does not give the intended interpretation of the Kleene star operator. To see this, consider the simple tree $t$ consisting of a root and two outgoing edges $a$ and $b$. Intuitively, the above definition would force all the branches of $t^*$ to be of the same depth. This also illustrates the underlying difference between our approach and that of standard dynamic logic: Sequential composition in our setting is defined over trees rather than over paths. The semantics of Kleene star, thus, needs to be defined with respect to a least fixed-point operator. We formalize this as follows: Let $\cdot$ be a binary operator over $W \times 2^W$, which is defined as:

- $R_1 \cdot R_2 = \{(u, X) \mid \exists w_1, Y_1, \dots, w_k, Y_k$ with $(u, \{w_1, \dots, w_k\}) \in R_1, \forall j, (w_j, Y_j) \in R_2$ and $X = \bigcup_j Y_j\}$.

for all $R_1, R_2 \subseteq W \times 2^W$.

Given a $Z \subseteq W \times 2^W$, let $F_Z$ be the operator over the domain $W \times 2^W$ defined as $F_Z(R) = R_\top \cup Z \cdot R$ where $R_\top = \{(u, \{u\}) \mid u \in W\}$. Observe that the operator $\cdot$ is monotonic in the following sense: If $R_1 \subseteq R_2$, then $R_0 \cdot R_1 \subseteq R_0 \cdot R_2$. This also implies that $F_Z$ is monotonic for every $Z \subseteq W \times 2^W$. Thus, by the Knaster–Tarski theorem we have that for every $Z$, the least fixed point ($\mathit{LFP}$) of $F_Z$ exists. $\mathit{LFP}(F_Z)$ can be computed as the limit of the following sequence of partial solutions: $R_0 = R_\top$, $R_{j+1} = F_Z(R_j)$ $(= R_\top \cup Z \cdot R_i)$, and $R_\lambda = \cup_{\nu < \lambda} R_\lambda$ for a limit ordinal $\lambda$. For $\mathcal{Q} \in \{\forall, \Box, \Diamond\}$, we define:

- $R^{\mathcal{Q}}_{\pi^*} = \mathit{LFP}(F_{R^{\mathcal{Q}}_{\pi}})$.

We are now in a position to formally define truth in a model:

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

DEFINITION 4.9 (Truth). *The* **truth** *of a formula $\alpha \in \mathcal{L}_{EPL}$ in a model $M = (W, \rightarrow, \rightsquigarrow, V)$ at a position $u$ (denoted $M, u \models \alpha$) is defined as follows:*

- $M, u \models p$ *iff* $p \in V(u)$
- $M, u \models \neg\alpha$ *iff* $M, u \not\models \alpha$
- $M, u \models \alpha_1 \vee \alpha_2$ *iff* $M, u \models \alpha_1$ *or* $M, u \models \alpha_2$
- $M, u \models \Box\alpha$ *iff for all* $w$ *such that* $u \rightsquigarrow w$ *we have* $M, w \models \alpha$
- $M, u \models \langle\pi\rangle^{\exists}\alpha$ *iff* $\exists(u, X) \in R_\pi^{\exists}$, $\exists w \in X$ *such that* $M, w \models \alpha$
- $M, u \models \langle\pi\rangle^{\forall}\alpha$ *iff* $\exists(u, X) \in R_\pi^{\forall}$ *such that* $\forall w \in X$ *we have* $M, w \models \alpha$
- $M, u \models \langle\pi\rangle^{\Box}\alpha$ *iff* $\exists(u, X) \in R_\pi^{\Box}$ *such that* $\forall w \in X$ *we have* $M, w \models \alpha$
- $M, u \models \langle\pi\rangle^{\Diamond}\alpha$ *iff* $\exists(u, X) \in R_\pi^{\Diamond}$ *such that* $\forall w \in X$ *we have* $M, w \models \alpha$

*where for $\mathcal{Q} \in \{\exists, \forall, \Box, \Diamond\}$, $R_\pi^{\mathcal{Q}} \subseteq W \times 2^W$ is defined above. The logical notions satisfiability and validity are defined as usual.*

The first technical contribution of this paper is a sound and (weakly) complete axiom system (in the language $\mathcal{L}_{EPL}$) for the class of all arenas with imperfect information. A straightforward consequence of this completeness proof is decidability of the satisfiability problem, which we discuss below.

The axiomatization and completeness proof extends the one found in Ramanujam & Simon (2009) to situations with imperfect information. In this section, we present this axiom system and discuss the proof (details can be found in Appendix A). First of all, note that the language $\mathcal{L}_{EPL}$ extends the standard PDL language: Let $e_a$ denote the tree $e_a = (x, a, y)$ with a single $a$-edge, and define for each $a \in \Sigma$, $\langle a\rangle\alpha = \langle e_a\rangle^{\exists}\alpha$. Given the semantics defined above (Definitions 4.8 and 4.9), we have the standard interpretation for $\langle a\rangle\alpha$: $\langle a\rangle\alpha$ holds at a state $u$ iff there is a state $w$ such that $u \xrightarrow{a} w$ and $\alpha$ holds at $w$.

A key observation is that whether a protocol $t$ is (subjectively) enabled can be described by a standard PDL formula. Formally, for each protocol $T$, let $t^{\surd}$ be a formula that is intended to denote that the tree structure $t$ is enabled. This is defined inductively on the structure of $t$ as:

- if $t = (x)$, then $t^{\surd} = \top$.
- if $t = (x, a_1, t_{a_1}) + \ldots + (x, a_k, t_{a_k})$, then

$$t^{\surd} = (\textstyle\bigwedge_{j=1,\ldots,k}(\langle a_j\rangle\top \wedge [a_j]t_{a_j}^{\surd})).$$

We use the formula $t^{\Box\surd}$ to denote that the protocol $t$ is subjectively enabled:

- if $t = (x)$, then $t^{\Box\surd} = \top$.
- if $t = (x, a_1, t_{a_1}) + \ldots + (x, a_k, t_{a_k})$, then

$$t^{\Box\surd} = (\textstyle\bigwedge_{j=1,\ldots,k}(\Box\langle a_j\rangle\top \wedge \Box[a_j]t_{a_j}^{\Box\surd})).$$

It is straightforward to check that these definitions work as intended:

LEMMA 4.10. *For any protocol $T$ and model $M = (W, \rightarrow, \rightsquigarrow, V)$, for each $w \in W$, $M, w \models t^{\surd}$ iff enabled$(t, w)$ holds, and $M, w \models t^{\Box\surd}$ iff s-enabled$((\mathcal{G}, w), t)$ holds.*

The above reductions from trees to standard PDL formulas suggest that the methods of Kozen & Parikh (1981) to prove completeness of PDL are also applicable in our setting. Our axiomatization follows this "reduction axiom" methodology (i.e., the Segerberg axioms for complex programs) with one important twist: Since the atomic protocols still encode the structure of a tree, we need to provide "reduction axioms" for atomic protocol

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

trees as well. The key idea is to define a formula $push_{\mathcal{Q}}(t, \alpha)$ for $\mathcal{Q} \in \{\exists, \forall, \Box\}$ which means that $t$ is (subjectively) enabled and that $\alpha$ holds at all the frontier nodes selected by the relation $R_t^{\mathcal{Q}}$. These formulas will be defined by induction on the structure of $t$: For atomic trees $t = (x)$,

(C1) $push_{\exists}((x), \alpha) = \alpha$.
(C2) $push_{\forall}((x), \alpha) = \alpha$.
(C3) $push_{\Box}((x), \alpha) = \Box\alpha$.

For $t = (x, a_1, t_{a_1}) + \ldots + (x, a_k, t_{a_k})$ and $A = \{a_1, \ldots, a_k\}$, we have

(C4) $push_{\exists}(t, \alpha) = \bigvee_{a_m \in A} \langle a_m \rangle \langle t_{a_m} \rangle^{\exists} \alpha$.
(C5) $push_{\forall}(t, \alpha) = \bigwedge_{a_m \in A} [a_m] \langle t_{a_m} \rangle^{\forall} \alpha$.
(C6) $push_{\Box}(t, \alpha) = \bigwedge_{a_m \in A} \Box[a_j] \langle t_{a_j} \rangle^{\Box} \alpha$.

Note that we have not given the corresponding formula for $\langle t \rangle^{\Diamond} \alpha$. This formula is of a different nature than the formulas above. The intended interpretation of $\langle t \rangle^{\Diamond} \alpha$ is that the protocol $t$ is subjectively enabled and $\alpha$ holds at all frontier nodes reached along a *subjective path* in $t$. Formally, (recall that $Paths(t)$ is the set of maximal paths in $T_t$), when the path consists of a single node (i.e., $\rho = (x)$) we have:

(P1) $cpath((x), \alpha) = \Box\alpha$.

When the path $\rho$ is consists of at least two nodes, we have:

(P2) $cpath(\rho, \alpha) = \Box[head(\rho)] cpath(tail(\rho), \alpha)$.

DEFINITION 4.11 (Axiomatization). *The* **epistemic protocol logic**, *which we denote* EPL, *is the smallest set of formulas from* $\mathcal{L}_{EPL}$ *containing all instances of the following axiom schemes and closed under the following inference rules:*

**Propositional Tautologies**

*(A1) All instances of propositional tautologies.*

**Normality Axioms**

*(A2)* *(a)* $\langle \pi \rangle^{\exists}(\alpha_1 \vee \alpha_2) \equiv \langle \pi \rangle^{\exists} \alpha_1 \vee \langle \pi \rangle^{\exists} \alpha_2$
*(b)* $\Box\alpha_1 \wedge \Box(\alpha_1 \supset \alpha_2) \supset \Box\alpha_2$

**Reduction axioms for atomic and composite protocols**

*(A3)* $\langle t \rangle^{\forall} \alpha \equiv t^{\surd} \wedge push_{\forall}(t, \alpha)$
*(A4)* $\langle t \rangle^{\exists} \alpha \equiv t^{\surd} \wedge push_{\exists}(t, \alpha)$
*(A5)* $\langle t \rangle^{\Box} \alpha \equiv t^{\Box\surd} \wedge push_{\Box}(t, \alpha)$
*(A6)* $\langle t \rangle^{\Diamond} \alpha \equiv t^{\Box\surd} \wedge \bigvee_{\rho \in Paths(t)} cpath(\rho, \alpha)$ *for* $\mathcal{Q} \in \{\exists, \forall, \Box, \Diamond\}$
*(A7)* $\langle \pi_1 \cup \pi_2 \rangle^{\mathcal{Q}} \alpha \equiv \langle \pi_1 \rangle^{\mathcal{Q}} \alpha \vee \langle \pi_2 \rangle^{\mathcal{Q}} \alpha$
*(A8)* $\langle \pi_1 ; \pi_2 \rangle^{\mathcal{Q}} \alpha \equiv \langle \pi_1 \rangle^{\mathcal{Q}} \langle \pi_2 \rangle^{\mathcal{Q}} \alpha$
*(A9)* $\langle \pi^* \rangle^{\mathcal{Q}} \alpha \equiv \alpha \vee \langle \pi \rangle^{\mathcal{Q}} \langle \pi^* \rangle^{\mathcal{Q}} \alpha$

**Inference rules**

$$(MP)\ \frac{\alpha, \alpha \supset \beta}{\beta} \quad (ANec)\ \frac{\alpha}{[a]\alpha} \quad (KNec)\ \frac{\alpha}{\Box\alpha}$$

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

$$(IND_{\mathcal{Q}}) \quad \frac{\langle \pi \rangle^{\mathcal{Q}} \alpha \supset \alpha}{\langle \pi^* \rangle^{\mathcal{Q}} \alpha \supset \alpha} \qquad \text{for } \mathcal{Q} \in \{\exists, \forall, \Box, \Diamond\}$$

Some remarks are in order. First, restricting attention to *finite* trees ensures that that the disjunction in axiom A6 is finite. Second, note that normality axioms for $\langle \pi \rangle^{\forall}$ and $\langle \pi \rangle^{\Box}$ are *not* valid. Finally, since the action modalities make assertions about the frontier of trees (and forests), the relation $R_{\pi}^{\mathcal{Q}}$ is not "upward closed." Nonetheless, the usual PDL axiom for composite programs is still sound:

PROPOSITION 4.12. *$\langle \pi_1; \pi_2 \rangle^{\mathcal{Q}} \alpha \equiv \langle \pi_1 \rangle^{\mathcal{Q}} \langle \pi_2 \rangle^{\mathcal{Q}} \alpha$ is valid for $\mathcal{Q} \in \{\exists, \forall, \Box, \Diamond\}$.*

*Proof.* We give a proof for the case when $\mathcal{Q} = \forall$, the other cases are similar. Suppose that $M, u \models \langle \pi_1; \pi_2 \rangle^{\forall} \alpha$. We will show $M, u \models \langle \pi_1 \rangle^{\forall} \langle \pi_2 \rangle^{\forall} \alpha$. Since $M, u \models \langle \pi_1; \pi_2 \rangle^{\forall}$, there exists $(u, X) \in R^{\forall}_{\pi_1;\pi_2}$ such that $\forall w \in X$, $M, w \models \alpha$. Hence, there exists $Y = \{v_1, \ldots, v_k\}$ such that $(u, Y) \in R^{\forall}_{\pi_1}$ and $\forall v_j \in Y$, there exists $X_j \subseteq X$ such that $(v_j, X_j) \in R^{\forall}_{\pi_2}$ and $\bigcup_{j=1,\ldots,k} X_j = X$. Therefore, $\forall v_k \in Y$, we have $M, v_k \models \langle \pi_2 \rangle^{\forall} \alpha$ and, hence, $M, u \models \langle \pi_1 \rangle^{\forall} \langle \pi_2 \rangle^{\forall} \alpha$.

Conversely, suppose that $M, u \models \langle \pi_1 \rangle^{\forall} \langle \pi_2 \rangle^{\forall} \alpha$. We will show $M, u \models \langle \pi_1; \pi_2 \rangle^{\forall} \alpha$. We have $M, u \models \langle \pi_1 \rangle^{\forall} \langle \pi_2 \rangle^{\forall} \alpha$ iff there exists $(u, Y) \in R^{\forall}_{\pi_1}$ such that $\forall v_k \in Y$, $M, v_k \models \langle \pi_2 \rangle^{\forall} \alpha$. $M, v_k \models \langle \pi_2 \rangle^{\forall} \alpha$ iff there exists $(v_k, X_k) \in R^{\forall}_{\pi_2}$ such that $\forall w_k \in X_k$, $M, w_k \models \alpha$. Let $X = \bigcup_k X_k$; from the definition of $R^{\forall}$ we get $(u, X) \in R^{\forall}_{\pi_1;\pi_2}$. Hence, $M, u \models \langle \pi_1; \pi_2 \rangle^{\forall} \alpha$. □

We can now state the two main theorems of this section:

THEOREM 4.13. EPL *is sound and weakly complete with respect to the class of all arenas with imperfect information.*

The proof of this theorem is found in Appendix A.

COROLLARY 4.14. *The satisfiability problem for* EPL *is decidable in nondeterministic double exponential time.*[17]

REMARK 4.15. *Note that the definition of subjectively enabled considers only single steps of the $\rightsquigarrow$ relation. One natural generalization here is to consider the transitive closure of $\rightsquigarrow$ in Definition 3.3. This suggests extending the language with a $\Box^*$ operator, which in turn may open the door to the many axiomatization issues in epistemic temporal languages with common knowledge (cf. van Benthem & Pacuit, 2006, for references and a discussion). Also relevant here are the axiomatizations of products of PDL and various epistemic and doxastic logics (Schmidt & Tishkovsky, 2008).*

We can incorporate the properties discussed in Section §3. Recall that a formula $\phi \in \mathcal{L}_{EPL}$ is valid in an arena (with imperfect information) if it is valid in every model based on the arena. First, note that a standard modal *correspondence* argument (cf. Blackburn *et al.*, 2002, chap. 3) gives us:

[17] This is an upper bound; the precise lower bound of the satisfiability problem is left open. The proof is a direct consequence of the proof of the completeness theorem since we construct a finite model.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

LEMMA 4.16. *Let* $\mathcal{G}^I = (W, \{\rightarrow_a\}_{a\in\Sigma}, \rightsquigarrow)$ *be an arena with imperfect information. Then,*

- $\mathcal{G}^I$ *satisfies no miracles iff* $[a]\Box\alpha \supset \Box[a]\alpha$ *is valid.*
- $\mathcal{G}^I$ *satisfies success iff* $\Diamond\langle a\rangle\top \supset \langle a\rangle\top$ *is valid.*
- $\mathcal{G}^I$ *satisfies awareness iff* $\langle a\rangle\top \supset \Box\langle a\rangle\top$ *is valid.*
- $\mathcal{G}^I$ *satisfies certainty of actions iff* $\Diamond\langle a\rangle\top \supset \Box\langle a\rangle\top$ *is valid.*

Furthermore, it is not hard to see that adding the axioms in the above lemma to the axioms in Definition 4.11 leads to a sound and weakly complete axiomatization of the relevant class of models.

**§5. Joint protocols.** The central issue addressed in this paper is the circumstances under which an agent can "knowingly" agree to follow a protocol or plan. We have seen that even in the single-agent case, this notion is interesting and nontrivial to formalize. However, the situation becomes even more interesting and complex in situations with more than one agent. A first approach to the multiagent situation is to assume that each agent follows her own "local" protocol. More formally, we can associate with each agent a set of local actions and define (local) protocols for each agent as before (Definitions 2.2 and 4.6) based on the agents' local actions. This is the underlying idea behind the *interpreted systems* of Halpern and others (see Fagin *et al*., 1995, for a discussion and references to the relevant literature). However, this approach hides an important distinction between an action *profile* and a *joint* action. The former is a sequence of (individual) actions describing choices made by each agent, while the latter involves an additional component "gluing" the agents' actions together.

The nature of this additional component is the subject of much debate among philosophers (cf. Bratman, 1999, on "shared intentions," Gilbert, 1989, on "joint commitments," and Sugden, 2003, on "team reasoning," among others Searle, 1990; Tuomela, 2010). The logic we present in this section does not commit to any specific view of joint actions. Our goal is to extend the analysis from the previous sections to the many-agent situation where some of the basic actions are classified as joint actions. Taking inspiration from concurrency theory, we use a "location function" that specifies for each basic action a set of agents *involved in the action*. Formally, the function $agents : \Sigma \rightarrow 2^N$ specifies for each action the subset of players associated with the action. Such a function has been extensively used in the analysis of asynchronous systems, where it typically specifies synchronized communication among a group of agents (Zielonka, 1987; Mukund & Sohoni, 1997). Also relevant for this paper is Ramanujam, 1996, where such a function is used to explicitly specify the source of agents' uncertainty in terms of synchronization actions. In this paper, if $i \in agents(a)$, then this means that agent $i$ is involved in the execution of $a$. In other words, in order to do action $a$, each agent in $agents(a)$ must do his or her "part" (whatever that may be). For example, if $a$ is the action "lift the piano" and $agents(a) = \{i, j\}$, then doing action $a$ means that $i$ lifts the left side and $j$ lifts the right side (or vice versa). Thus, we do not specify *how* the agents in $agents(a)$ go about doing action $a$.

The definition of a (basic) protocol is the same as in the previous sections. That is, a (joint) protocol is a finite labeled tree (Definition 2.2) where the labels now represent joint actions. While this move to the many-agent setting raises many technical and conceptual questions (cf. Lorini *et al*., 2009), we focus on one specific question: What does it means for a joint protocol to be subjectively enabled? In the single-agent case this was defined by taking the closure under the uncertainty relation of the player at every stage (or node

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

corresponding to the protocol tree). This technique does not work in the case of joint protocols since the actions (and not the nodes) specify the agents involved in the protocol. Below, we discuss a number of ways to solve this problem. We start with a very simple solution: Whenever a joint action $a$ is in a protocol, we require that for each agent $i \in agents(a)$, the action $a$ must be enabled at all the states that $i$ considers possible (specified by $i$'s uncertainty relation). We formalize these ideas below.

*Many-agent epistemic protocol logic.* Let $N = \{1, \ldots, n\}$ be the set of agents. A multiagent arena with imperfect information is a tuple $\mathcal{G}^I = (W, \{\rightarrow_a\}_{a\in\Sigma}, \{\rightsquigarrow_i\}_{i\in N})$ where $(W, \{\rightarrow_a\}_{a\in\Sigma})$ is a finite arena, as earlier, and for each $i \in N$, the relation $\rightsquigarrow_i \subseteq W \times W$ specifies the uncertainty of agent $i$. For a position $u \in W$, let $\mathcal{I}_i(u) = \{w \in W \mid u \rightsquigarrow_i w\}$. The definition of a protocol remains the same as earlier: a (finite) labeled tree where the labels are now interpreted as joint actions. For a protocol tree $T$, let $\xi_T$ denote the set of agents involved in the protocol $T$—that is, $\xi_T = \{i \in N \mid \exists s \in S, \exists a \in \mathcal{A}(s)$ with $i \in agents(a)\}$. Thus, a *local protocol* for agent $i$ is one in which $\xi_T = \{i\}$ or, in other words, for all actions $a$ occurring in $T$ we have $agents(a) = \{i\}$. For any finite path $\rho$ such that $\mathcal{A}(last(\rho)) \neq \emptyset$, let $\mathcal{N}(\rho) = \{i \in N \mid \exists a \in \mathcal{A}(last(\rho))$ and $i \in agents(a)\}$.

The **subjective restriction** of $T = (S, \{\Rightarrow_a\}_{a\in\Sigma}, s_0)$ in $(\mathcal{G}^I, u)$ (which we denote $(\mathcal{G}^I, u) \restriction_s T$) is a tuple $(X, f, act)$ where $X$ is a forest in $\mathcal{G}^I$, $f$ is a map $f : X \rightarrow S$ and $act$ is a map $act : X \rightarrow 2^\Sigma$. This is defined inductively as follows:

0. We have two cases to consider:
   - if $\mathcal{A}(s_0) = \emptyset$, then $X_0 = \bigcup_{i\in\xi_T} \mathcal{I}_i(u)$; and
   - if $\mathcal{A}(s_0) \neq \emptyset$, then $X_0 = \bigcup_{i\in\mathcal{N}(s_0)} \mathcal{I}_i(u)$ ($v \in X_0$ is understood as a one-element sequence). For all $v \in X_0$, set $f_0(v) = s_0$ and $act(v) = \{a \in \mathcal{A}(s_0) \mid v \in \mathcal{I}_i(u)$ and $i \in agents(a)\}$.

n. Suppose that $X_n$ and $f_n$ have been constructed. For each $\rho \in S_n$, we need to consider two cases:
   - if $\mathcal{A}(f_n(\rho)) \neq \emptyset$, then for all $a \in act(\rho)$, let
   $$Y_a^\rho = \{\rho a w \mid last(\rho) \rightarrow_a w \text{ in } \mathcal{G}^I\} \text{ and } Z_a^\rho = \bigcup_{\rho'\in Y_a^\rho} \bigcup_{i\in\mathcal{N}(\rho')} \mathcal{I}_i(\rho').$$
   Let $f_{n+1}$ extend $f_n$ such that for all $\rho' \in Z_a^\rho$, $f_{n+1}(\rho') = s'$ where $f_n(\rho) \Rightarrow_a s'$ in $T$. The map $act_{n+1}$ is also defined as an extension of $act_n$, where for all $\rho' \in Z_a^\rho$ we define $act_{n+1}(\rho') = \{a \in \mathcal{A}(last(\rho)) \mid last(\rho') \in \mathcal{I}_i(\rho)$ and $i \in agents(a)\}$. Define
   $$X_{n+1} = X_n \cup \bigcup_{\rho\in X_n, a\in\mathcal{A}(f_n(\rho))} Z_a^\rho.$$
   - if $\mathcal{A}(f_n(\rho)) = \emptyset$, then $X_{n+1} = X_n \cup \bigcup_{\rho\in X_n, i\in\xi_T} \mathcal{I}_i(\rho)$.

Finally, let $X = X_{height(T)}$, $f = f_{height(T)}$ and $act = act_{height(T)}$.

REMARK 5.17. *Observe that in the above definition, when $\mathcal{A}(f_k(\rho)) = \emptyset$ for some $k$, we take the closure under the uncertainty relation of all the agents involved in the protocol $T$. This reflects the fact that at the end of the protocol, the analysis takes into account the uncertainty of all the players involved in the protocol. In general, one could consider any group of agents $\xi \subseteq N$ and define the closure with respect to this group.*

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

DEFINITION 5.18 (Subjectively Enabled for Joint Protocols). *A joint protocol $T$ is* **subjectively enabled** *at $u$ in $\mathcal{G}^I = (W, \rightarrow, \rightsquigarrow)$ if the structure $(\mathcal{G}^I, u) \upharpoonright_s T = (X, f, act)$ satisfies the condition $\forall \rho \in X$, $act(\rho) = \mathcal{A}(\rho)$.*

Thus, in order for a joint protocol to be subjectively enabled, it is required that for each joint action $a$ in the protocol, for all $i \in agents(a)$, $a$ is enabled at all the states that $i$ considers possible. Of course, this is only one of many different ways to formally define what it means for a joint protocol to be subjectively enabled. Another approach would be to require that the relevant actions are enabled in the states in the *intersection* of the uncertainty relation of the agents involved. This corresponds to the *distributed knowledge* of the relevant agents. At the other extreme, we could base our definition of subjectively enabled on the *common knowledge* of the relevant agents. A detailed analysis of this and other issues raised by the many-agent setting will be left for future work. We conclude this section with a brief discussion of axiomatic issues.

Given the above definition of subjectively enabled, we can prove a completeness theorem for the class of multiagent arenas with imperfect information (in the obvious language) using the methods discussed in Section 4.1. The crucial observation is that a joint protocol being subjectively enabled is expressible in a multiagent epistemic PDL language. Given a protocol specification $t$, let $\xi_t$ be the set of agents involved in $t$—that is, $\xi_t = \{i \in N \mid \exists s \in T_t, \exists a \in \mathcal{A}(s)$ such that $i \in agents(a)\}$. The formula $t^{\Box\surd}(\xi_t)$, which denotes that the joint protocol $t$ is subjectively enabled, is defined as:

- if $t = (x)$, then $t^{\Box\surd}(\xi_t) = \bigwedge_{i \in \xi_t} \Box_i \top$.
- if $t = (x, a_1, t_{a_1}) + \ldots + (x, a_k, t_{a_k})$ with $A = \{a_1, \ldots, a_k\}$, then

$$t^{\Box\surd}(\xi_t) = \bigwedge_{a_j \in A} \bigwedge_{i \in agents(a_j)} (\Box_i \langle a_j \rangle \top \wedge \Box_i [a_j] t_{a_j}^{\Box\surd}(\xi_t)).$$

**§6. Conclusion and discussion.** This paper focuses on the interplay between epistemic reasoning and protocol analysis. In particular, we developed an epistemic protocol logic and discussed what it means for an agent to "subjectively" agree to follow a given protocol. We see this as one step towards addressing the fundamental problem of how to model agents "knowing a protocol, plan, or strategy" in situations with imperfect information, and we proved a number of results about our logical system. We conclude with a discussion of related and future work.

***6.1. Actions, abilities, and know-how.*** Our paper touches on a number of issues surrounding the relationship between an agent's abilities, available choices, and information in an interactive situation. The issues here are subtle and a complete discussion is beyond the scope of this paper; however, we would like to explain how our logical frameworks fit into to this broader literature. We assume that the agents may be uncertain about which (basic) actions are available (i.e., which choices are *feasible*). Amidst this uncertainty, the agents commit to a (joint) plan or protocol. Some features of our notion of a plan are worth highlighting:

- Plans are compositional: complex plans are built from simpler ones using the standard regular operators (concatenation, union, and Kleene star).
- Plans may be partial: basic protocols may be branching.
- We do not include tests in our language.

In this paper, we focus on the question *under what circumstances can an agent commit to a (joint) protocol or plan, and what can she achieve by doing so*? But, this is only one of

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

many different questions that can be investigated. We mention here three questions that are related to issues that have come up in this paper.

*What does it mean for an agent to "know a protocol"?* As we remarked in the Introduction, a common assumption is that it is *common knowledge* that there is a fixed protocol which all the agents have (implicitly or explicitly) agreed to follow. In what sense do the agents *know* the protocol? Formally, the protocol describes which states or histories are "in the model," so the *proposition* expressing that "the protocol is being followed" is the set of *all* elements in the model (i.e., the set $W$ of all possible worlds in the model). Thus, in terms of the agents' *propositional knowledge*, "knowing the protocol" amounts to knowing "that the set of possible states is $W$," but this just means that the agent knows that '$\top$'. Nonetheless, "knowing the protocol" has important practical and pragmatic ramifications on the agents' information. First, the protocol explicitly limits the available observations, messages, and/or actions available (or permitted) to the agent.[18] Second, the protocol affects how the agents interpret their observations (Parikh & Ramanujam, 2003). These two aspects of knowing a protocol are extensively discussed in Yanjing Wang's recent PhD thesis (Wang, 2010).[19]

*How do the agents come to know a protocol?* Our logical frameworks focus on what agents *can* achieve by committing to a protocol or plan. But, we do not address the *dynamics* of these commitments. A dynamic (epistemic) protocol logic has recently been introduced by Wang (2010, chap. 4). The key idea is to extend PDL with a program announcement modality, denoted $[!\pi]$ where $\pi$ is a PDL action expression (unlike us, Wang does not allow branching for atomic programs). Formulas are interpreted in the usual PDL models at a state *and a program expression* representing the protocol the agents are currently committed to. So, for example, it may be currently true that the agent can do $a$ (i.e., $a$ complies with the current protocol), but after announcing that the protocol is $b$, then $a$ is no longer compliant (this is represented by the formula $\langle a\rangle\top \wedge [!b]\neg\langle a\rangle\top$). This type of dynamics also makes sense in our setting. Indeed, it would be a very interesting line of research to add Wang's "protocol announcement" operators to our epistemic protocol logic (Section 4.1).

*What is the (formal) difference between an agent knowing that she can achieve $\phi$ and knowing how to achieve $\phi$?*[20] Much of the work on epistemic extensions of logics of actions and abilities has focused on the distinction between *de re*/*de dicto* knowledge of what agents can achieve (van Benthem, 2001b; Herzig & Troquard, 2006; van der Hoek & Wooldridge, 2003a; Jamroga & Agotnes, 2007). To illustrate the issue, we use an example from Herzig & Troquard (2006): Suppose that Ann, who is blind, is standing with her hand on a light switch. She currently does not know whether the light is on or off. The question is does she have the *ability* to turn the light on? Is she *capable* of turning the

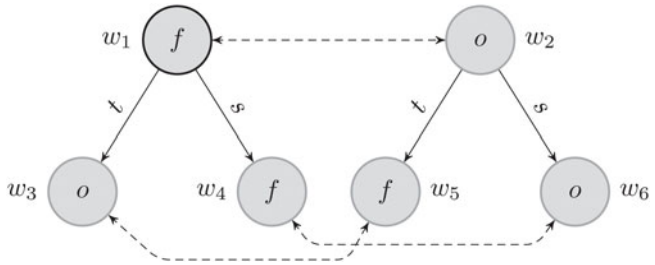
[18] So, for example, truth of $\phi$ no longer implies that $\phi$ can be announced (cf. van Benthem *et al*., 2009).

[19] In particular, see chapters 2 and 3.

[20] Philosophers since Ryle, 1949, chap. 2, have discussed the distinction between "knowing that" and "knowing how." Consult Fantl (2008) for an up-to-date survey of the current philosophical debate. Certainly, some of the issues raised in this debate are relevant to the discussion here, but we leave a complete analysis for a different occasion. See Carr (1979) and Singh (1999) for logical analyses of "knowing how" that is related to the framework we develop in this paper.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

light on? Does she *know how* to turn the light on?[21] We do not address these conceptual questions here, but, rather, illustrate some distinction we can make in our logical system. Ann has two options available to her: toggle the switch ($t$) or do nothing ($s$). This situation is represented by the following arena with imperfect information:




Suppose that the actual state is $w_1$, so the light is currently off. Now, since Ann is blind, she does not know that the light is off ($w_1 \models \neg\Box f$).[22] Furthermore, the following formulas are true at $w_1$: $\langle t\rangle o$ ("after toggling the light switch ($t$), the light will be on ($o$)"), $\neg\Box\langle t\rangle o$ ("Ann does not know that after toggling the light switch, the light will be on"), $\Box(\langle t\rangle\top \wedge \langle s\rangle\top)$ ("Ann knows that she can toggle the switch ($t$) and she can do nothing ($s$)," and $\langle t\rangle\neg\Box o$ ("after toggling the switch Ann does not know that the light is on"). These formulas describe the basic options available at $w_1$ and the information Ann has about these options. Consider the basic plan "turn the light on"[23] (denoted by $l$). Agreeing to this plan commits Ann to a choice between $t$ and $o$, but this choice can only be made "in the moment" (since, the "correct" option depends on the state of affairs). So, $l$ is a basic protocol consisting of a tree with two branches, one labeled with $t$ and the other labeled with $o$. We have:

- $w_1 \models \langle l\rangle^{\exists} o \wedge \neg\langle l\rangle^{\forall} o$: executing the plan "turning the light on" can lead to a situation where the light is on, but this is not *guaranteed* (the plan may fail).
- $w_1 \models \Box\langle l\rangle^{\exists} o$: Ann knows that she is capable of turning the light on. She has *de dicto* knowledge that she can turn the light on.
- $w_1 \models \neg\langle l\rangle^{\Diamond} o$: Ann cannot knowingly turn on the light (she does not have *de re* knowledge that she can turn the light on): there is no *subjective* path leading to states satisfying $o$ (note that *all* elements of the last element of the subject path must satisfy $o$).[24]

So, our logical framework can express interesting relationships between a plan $\pi$, propositions that can be "brought about" by following $\pi$, and what the agent(s) knows about $\pi$: For example, $\Box\langle\pi\rangle^{\forall}\phi$ means "the agent *knows that* she can bring about $\phi$ by following $\pi$," $\langle\pi\rangle^{\forall}\Box\phi$ means "the agent *can* bring about her knowledge of $\phi$ by following $\pi$," and $\langle\pi\rangle^{\Box}\phi$

[21] There is an interesting philosophical debate about what are the right distinctions to make here (cf. Lesperance *et al*., 2000, for a perspective from artificial intelligence) and the precise relationship with *propositional knowledge* (cf. Stanley & Williamson, 2001; Williams, 2008). In the interest of space, we do not discuss these interesting issues here.

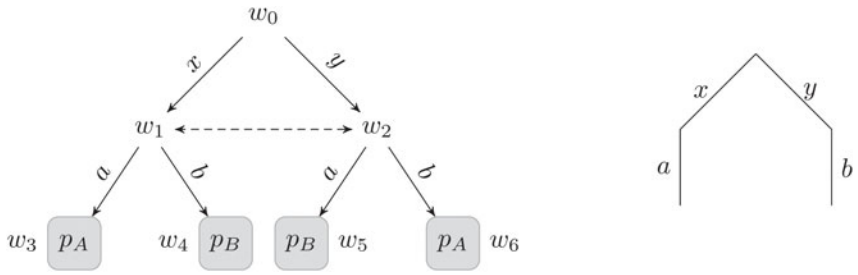
[22] We do not label the modal operator since Ann is the only agent.

[23] Alternatively, we may use the command "make sure the light is on!" for this plan.

[24] It is interesting to note that if $t$ was informative for Ann, so that there is no uncertainty for Ann between states $w_3$ and $w_5$, then $\langle l\rangle^{\Diamond} o$ would be true at state $w_1$. For example, suppose that Ann was not blind, but was standing outside of the room with the door shut and $t$ was the action "open the door."

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

means "the agent *knows how* to follow $\pi$ in order to bring about $\phi$." Arguably, the issues discussed above become even more pressing when developing logics of explicit strategies for reasoning about game-theoretic situations (van Benthem, 2008). In particular, a player may know that she can win the game without actually knowing how (see van Benthem, 2001b, for a discussion). We conclude this subsection with an initial discussion about how to use our framework for reasoning about strategies in games with imperfect information. Consider an extensive game where Bob moves first (choosing between $x$ and $y$) and Ann moves second (choosing between $a$ and $b$) without knowledge of Bob's choice:




Suppose that $p_A$ denotes a win for Ann and $p_B$ a win for Bob. Let $s$ be the plan on the right which can be thought of as a strategy for Ann. Indeed, this is a winning strategy for Ann: $w_0 \models \langle s \rangle^{\forall} p_A$. Furthermore, Ann knows that this is a winning strategy, $w_0 \models \Box \langle s \rangle^{\forall} p_A$ (assume that $w_0 \in \mathcal{I}(w_0)$ for Ann). However, even though this strategy is subjectively enabled for Ann, she does not know how to use this strategy to win the game (in the terminology of van Benthem, 2001b: the strategy is not *prescriptive*:[25] That is, we have $w_0 \models \neg \langle s \rangle^{\Box} p_A$. These are only some initial observations about how to use our logical framework to reason about strategies in imperfect information games—a complete discussion will be left for future work.

We conclude this section by observing that the definition of subjectively enabled (Definition 3.3) can be simplified when $\rightsquigarrow$ is an equivalence relation:

PROPOSITION 6.1. *Let $\mathcal{G}^I = (W, \{\rightarrow_a\}_{a \in \Sigma}, \rightsquigarrow)$ be an arena with imperfect information where $\rightsquigarrow$ is an equivalence relation that satisfies no miracles. Then, for any protocol $T$, $T$ is subjectively enabled at position $u$ in $\mathcal{G}^I$ iff there is a function $f$ mapping nodes in $T = (S, \{\Rightarrow_a\}_{a \in \Sigma}, s)$ to positions in $\mathcal{G}^I$ ($f : S \rightarrow W$) such that*

1. *$f(s) = u$; and*
2. *for all $t \in S$, if $a \in \mathcal{A}(t)$ and $v \in \mathcal{I}(f(t))$, then $a \in \mathcal{A}(v)$.*

This simple (but instructive) proof is left to the reader. This proposition is important because it can be used to establish connections between our work and existing literature on related topics. Much of the current work on protocols and strategies discusses epistemic issues: Witness the "knowledge programs" of Fagin *et al*. (1997) and Halpern & Fagin

[25] This should be contrasted with a strategy that is *uniform*. In our terminology, a protocol $\pi$ is *uniform* if it is subjectively enabled and it is prescriptive for $\phi$ if $\langle \pi \rangle^{\Box} \phi$ is true at the root node. van Benthem, 2001b, showed that in games with perfect recall a winning strategy for player $i$ is uniform iff it is prescriptive (for the proposition expressing that player $i$ won the game).

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

(1989) and the recent contributions of Jan van Eijck and Yanjing Wang, as well as others using PDL to reason about *executing* a knowledge program (van Eijk *et al*., 2009; Wang, 2010). The focus here tends to be on knowing some objective, under the assumption that the agents implicitly agree to follow a "knowledge protocol" designed by the modeler to achieve the objective. Our work suggests a different question where the protocols themselves can be the object of knowledge: Given some (epistemic) objective, is there a protocol that the agents can (knowingly) agree to follow that will achieve the objective? Certainly, much more can be said on this topic (and has: see, e.g., Lesperance *et al*., 2000; Herzig *et al*., 2000), but this will be left for future work.

***6.2. Comparisons.*** There are many other interesting questions to ask about the logical system introduced in the previous section. For example, we can show that $\mathcal{L}_{EPL}$ is strictly more expressive than the language of PDL (both interpreted over labelled transition systems), but what about concurrent PDL, game logic, the modal $\mu$-calculus, or branching time temporal logic (CTL)? This section contains a number of preliminary observations; a more detailed comparison with related logical systems will be left for future work. It is easy to see that PDL is a fragment of $\mathcal{L}_{EPL}$ (indeed, we use this in our axiomatization). Furthermore, a simple adaptation of Peleg's (1987) argument showing that concurrent PDL (CPDL) is strictly more expressive than PDL shows that $\mathcal{L}_{EPL}$ without $\langle\pi\rangle^{\Box}$ and $\langle\pi\rangle^{\Diamond}$ (as considered in Ramanujam & Simon, 2009) is strictly more expressive than PDL. The main idea stems from a crucial observation made in Peleg (1987): No PDL formula can express the property of spawning an unbounded number of processes in parallel. This can be expressed in $\mathcal{L}_{EPL}$ using the branching in our atomic programs.

**Observation 1** *$\mathcal{L}_{EPL}$ is strictly more expressive than PDL.*

*Proof.* Consider the formula: $\phi = \langle(t; (a \cup b))^*\rangle^{\forall}[a]\bot$ where $t = (x, a, y_1)+(x, b, y_2)$. In other words, $t$ is the tree with two branches labeled $a$ and $b$. Consider the model $M$ (see Peleg, 1987, p. 459, figure 1), consisting of an infinite sequence of states labeled 0,1,2, ... From every state $i$, $i > 0$ there are both $a$ and $b$ edges leading to $i - 1$. There is also a "bypassing" edge from every odd state $2i + 1$ to state $2i - 1$.

It can be easily verified that, on the one hand, $\phi$ is satisfiable in every even state of the model. On the other hand, no odd state satisfies $\phi$ since in any restriction starting from an odd state, some branch will always be forced to remain on the main path and will not use bypasses. Therefore, $\phi$ precisely describes the even states of the model. Now, from Peleg (1987, pp. 472–475), we have the following lemma:

LEMMA 6.2 (Peleg, 1987). *Every PDL formula in the model M defines either a finite or a cofinite set.*

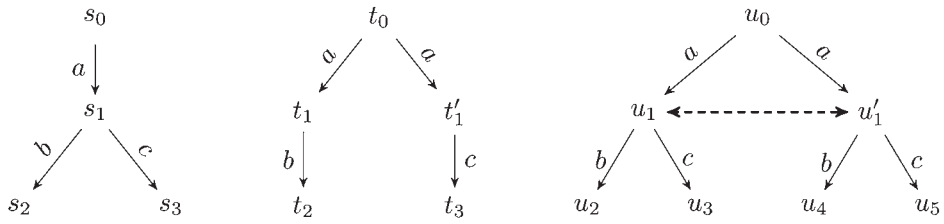
It follows that no PDL formula can be equivalent to $\phi$. □

Of course, the interesting question is whether our language $\mathcal{L}_{EPL}$ is more expressive than that of CPDL. Indeed, $\mathcal{L}_{EPL}$ is very similar to the language of CPDL. The crucial difference is that CPDL allows parallel branching on arbitrary programs: There is a program operator '$\pi_1 \cap \pi_2$' meaning "execute $\pi_1$ and $\pi_2$ in parallel." However, parallel branching occurs only at the atomic level in $\mathcal{L}_{EPL}$. Thus, determining whether $\mathcal{L}_{EPL}$ is as expressive as the language of CPDL reduces to showing that every regular expression involving a parallel operator $\cap$ can be rewritten as a regular expression where the atomic programs are finite trees. A related question is can we characterize the fragment of the $\mu$-calculus that

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

is equivalent to our epistemic protocol logic?[26] We leave these interesting questions for future work.

Finding the precise relationship between our epistemic protocol logic and other logical frameworks raises an important question: can we characterize the expressive power of our epistemic protocol language (over the class of arenas with imperfect information). In order to tackle this problem, we need a notion of equivalence between models corresponding to equivalence with respect to $\mathcal{L}_{EPL}$. For example, it is well known that (standard) PDL formulas are invariant under *bisimulation*,[27] and this fact is instrumental in helping us understand the precise relationship between PDL and other logical languages (interpreted over the same structures, such as the $\mu$-calculus). Consider the following arenas:




$\mathcal{G}_1$ $\mathcal{G}_2$ $\mathcal{G}_3$

Note that $\mathcal{G}_1$ and $\mathcal{G}_2$ are *trace equivalent*[28] but not bisimilar; and, indeed, it is not hard to find a formula of $\mathcal{L}_{EPL}$ that can distinguish these models.[29] Furthermore, it is not hard to see that there is no formula of $\mathcal{L}_{EPL}$ that can distinguish $\mathcal{G}_1$ and $\mathcal{G}_3$. An interesting line of research, which we leave for future work, is to find the appropriate notion of equivalence between models (cf. Nicola, 1987).

Another interesting question concerns the choice of the modal language. Note that the modalities '$\langle\pi\rangle^{\Box}$' and '$\langle\pi\rangle^{\Diamond}$' are "epistemized" versions of the action modalities '$\langle\pi\rangle^{\forall}$' and '$\langle\pi\rangle^{\exists}$'. A natural question is whether we can drop the former modalities in favor of a more expressive protocol language that incorporates uncertainty in the tree structure. More generally, we would like to construct the "actual" uncertainty an agent faces as a consistent product of uncertainty described in the model and uncertainty specified in a protocol. This is closer in spirit to the notion of product update used in *dynamic epistemic logic* (see van Benthem, 2010, for references).

The main idea here is to consider protocol trees, denoted by $\mathcal{P}_{\epsilon}(\mathcal{V})$ (cf. Definition 4.6), over an extended alphabet set $\Sigma_{\epsilon} = \Sigma \cup \{\epsilon\}$. The $\epsilon$ edges specify the uncertainty relation in the atomic protocol tree, and the notion of a protocol $t$ being enabled at a state $u$ can be

[26] Note that the key construction in this paper (iterating a finitely branching tree) can also be represented in the $\mu$-calculus: For example, $\mu p.\langle a\rangle p \wedge \langle b\rangle p$ defines the same tree at $t^*$where $t = (x, a, (x_1)) + (x, b, (x_2))$ (the atomic tree with two branches one labeled with $a$ and the other with $b$).

[27] We assume the reader is familiar with the notion of bisimulation. See Blackburn *et al*. (2002) for a detailed discussion.

[28] *Trace equivalent* means that the models contain the same sequence of actions: In this case, $\{ab, ac\}$.

[29] Consider the basic protocol $t$ which consists of one $a$-edge followed by tree with two branches, one labeled with $b$ and the other with $c$. This protocol is enabled in $\mathcal{G}_1$ (indeed, it is *isomorphic* to $\mathcal{G}_1$) but not in $\mathcal{G}_2$.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

defined in a manner similar to Definition 3.3. The idea is that the $\epsilon$ edges in the protocol tree match the silent transitions $\rightsquigarrow$ present in the model. Of particular interest is the subclass of protocol trees $\mathcal{P}_\epsilon^{\Box}(\mathcal{V}) \subseteq \mathcal{P}_\epsilon(\mathcal{V})$ where the labels on the path strictly alternate between $\epsilon$ and an action symbol in $\Sigma$, and the $\epsilon$ edge is never combined with $\Sigma$ in the branching structure.

PROPOSITION 6.3. *If the protocol tree includes uncertainty (as described above), then the* $\langle\pi\rangle^{\Box}$ *and* $\langle\pi\rangle^{\Diamond}$ *modalities are definable using* $\langle\pi\rangle^{\forall}$ *and* $\langle\pi\rangle^{\exists}$.

A formal statement[30] of this proposition and a sketch of the proof can be found in Appendix B. This proposition shows that being able to specify the uncertainty relation directly on the protocol tree gives rise to a more general framework.

**§7. Acknowledgements.** The authors would like to thank Horacio Arló-Costa, Valentin Goranko, Johan van Benthem, R. Ramanujam, and the participants at Dagstuhl Seminar 11101, *Reasoning about Interaction: From Game Theory to Logic and Back* and the Amsterdam *Workshop on Epistemic Modeling and Protocol Analysis* for many valuable comments. This research was completed while the first author was supported by an NWO Vidi grant (016.094.345).

**A Proof of Theorem 4.13.** To show completeness, we prove that every consistent formula is satisfiable. Let $\alpha_0$ be a consistent formula, and $CL(\alpha_0)$ denote the subformula closure of $\alpha_0$. In addition to the usual closure, we also require that

1. $\langle t\rangle^{\mathcal{Q}}\alpha \in CL(\alpha_0)$ implies $push_{\mathcal{Q}}(t,\alpha) \in CL(\alpha_0)$ for $\mathcal{Q} \in \{\exists, \forall, \Box\}$ and
2. $\langle t\rangle^{\Diamond}\alpha \in CL(\alpha_0)$ implies $\bigvee_{\rho\in Paths(t)} cpath(\rho,\alpha) \in CL(\alpha_0)$.

Let $\mathcal{AT}(\alpha_0)$ be the set of all maximal consistent subsets of $CL(\alpha_0)$, referred to as atoms. We use $u, w$ to range over the set of atoms. Each $u \in \mathcal{AT}(\alpha_0)$ is a finite set of formulas and we denote the conjunction of all formulas in $u$ by $\widehat{u}$. For a nonempty subset $X \subseteq \mathcal{AT}$, let $\widetilde{X} = \bigvee_{w\in X} \widehat{u}$. Define the transition relation on $\mathcal{AT}(\alpha_0)$ as follows: $u \xrightarrow{a} w$ iff $\widehat{u} \wedge \langle a\rangle\widehat{w}$ is consistent. We define the uncertainty relation as: $u \rightsquigarrow w$ iff $\widehat{u} \wedge \Diamond\widehat{w}$ is consistent. The valuation $V$ is defined as $V(w) = \{p \in P \mid p \in w\}$. The model is $M = (W, \rightarrow, \rightsquigarrow, V)$ where $W = \mathcal{AT}(\alpha_0)$. We also make use of the following notation, for $u \in W$ and an action $a \in \Sigma$, let $(u,a)_{\rightarrow} = \{w \mid u \xrightarrow{a} w\}$. The key observations are:

- For all $\langle\pi\rangle^{\exists}\alpha \in CL(\alpha_0)$ and for all $u \in W$, $\widehat{u} \wedge \langle\pi\rangle^{\exists}\alpha$ is consistent iff there exists $(u, X) \in R_\pi^{\exists}$ and $w \in X$ such that $\alpha \in w$.
- For $\mathfrak{F} \in \{\forall, \Box, \Diamond\}$, for all $\langle\pi\rangle^{\mathfrak{F}}\alpha \in CL(\alpha_0)$ and for all $u \in W$, $\widehat{u} \wedge \langle\alpha\rangle^{\mathfrak{F}}$ is consistent iff there exists $(u, X) \in R_\pi^{\mathfrak{F}}$ such that for all $w \in X$, $\alpha \in w$.

We present proofs for the cases $\langle\pi\rangle^{\exists}\alpha$ and $\langle\pi\rangle^{\Box}\alpha$. The arguments for the remaining cases are similar. The following lemma can be shown using standard modal logic techniques.

LEMMA 1.1. *For all* $u \in W$, *we have the following properties.*

- *if* $\widehat{u} \wedge \langle a\rangle\alpha$ *is consistent then there exists* $w$ *such that* $u \rightarrow_a w$ *and* $\widehat{w} \wedge \alpha$ *is consistent.*
- *if* $\widehat{u} \wedge [a]\alpha$ *is consistent then for all* $w$ *such that* $u \rightarrow_a w$ *we have* $\widehat{w} \wedge \alpha$ *is consistent.*

[30] Goranko & Jamroga (2004) make a similar observation in the context of epistemic extensions of alternating-time temporal logic.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

- *if* $\widehat{u} \wedge \Box\alpha$ *is consistent then for all* $w$ *such that* $u \rightsquigarrow w$ *we have* $\widehat{w} \wedge \alpha$ *is consistent.*
- *if* $\widehat{u} \wedge \Diamond\alpha$ *is consistent then there exists* $w$ *such that* $u \rightsquigarrow w$ *and* $\widehat{w} \wedge \alpha$ *is consistent.*

LEMMA 1.2. *For all* $t \in \mathcal{P}(\mathcal{V})$*, for all* $u, w \in W$*, if* $\widehat{u} \wedge \langle t \rangle^{\exists} \widehat{w}$ *is consistent then* $\exists X \subseteq W$ *such that* $(u, X) \in R_t^{\exists}$ *and* $\vdash \widehat{w} \supset \widetilde{X}$.

*Proof.* By induction on the structure of $t$.

- $t = (x)$: From axiom (A4) case (C1) we get $\langle (x) \rangle^{\exists} \alpha \equiv \alpha$. The lemma follows from this quite easily.
- $t = (x, a_1, t_{a_1}) + \ldots + (x, a_k, t_{a_k})$: Suppose $\widehat{u} \wedge \langle t \rangle^{\exists} \widehat{w}$ is consistent, from axiom (A4) we get $\widehat{u} \wedge t^{\surd}$ is consistent. Therefore there exists sets $Y_1, \ldots, Y_k$ such that $\forall j : 1 \leq j \leq l$, for all $v_j^l \in Y_j$ we have $u \xrightarrow{a_j} v_j^l$. From (A4) case (C1) we get $\widehat{u} \wedge (\bigvee_{a_j \in A} \langle a_m \rangle \langle t_{a_m} \rangle^{\exists} \widehat{w})$ is consistent. Therefore there exists $v_m^r$ such that $u \xrightarrow{a_m} v_m^r$ and $v_m^r \wedge \langle t_{a_m} \rangle^{\exists} \widehat{w}$ is consistent. By induction hypothesis, for all $j, l$ we get $\exists X_j^l$ such that $(v_j^l, X_j^l) \in R_{t_{a_j}}^{\exists}$ and there exists $X_m^r$ such that $(v_m^r, X_m^r) \in R_{t_{a_m}}^{\exists}$, $\vdash \widehat{w} \supset \widetilde{X_m^r}$. Let $X = \bigcup_{j=1,\ldots,k} \bigcup_{l=1,\ldots,|Y_j|} X_j^l$, from semantics we get $(u, X) \in R_t^{\exists}$. We also have $\vdash \widetilde{X_m^r} \supset \widetilde{X}$ and $\vdash \widehat{w} \supset \widetilde{X_m^r}$ and thus $\vdash \widehat{w} \supset \widetilde{X}$ as required.

□

The following two lemmas can be proved using standard techniques.

LEMMA 1.3. *For all* $\pi \in \Gamma$*, for all* $u, w \in W$*, if* $\widehat{u} \wedge \langle \pi \rangle^{\exists} \widehat{w}$ *is consistent then* $\exists X \subseteq W$ *such that* $(u, X) \in R_\pi^{\exists}$ *and* $\vdash \widehat{w} \supset \widetilde{X}$.

LEMMA 1.4. *For all* $\langle t \rangle^{\exists} \alpha \in CL(\alpha_0)$ *and for all* $u \in W$ *if there exists* $(u, X) \in R_t^{\exists}$ *and* $w \in X$ *such that* $\alpha \in w$ *then* $\widehat{u} \wedge \langle t \rangle^{\exists} \alpha$ *is consistent.*

LEMMA 1.5. *For all* $\langle \pi \rangle^{\exists} \alpha \in CL(\alpha_0)$ *and for all* $u \in W$, $\widehat{u} \wedge \langle \pi \rangle^{\exists} \alpha$ *is consistent iff* $\exists (u, X) \in R_\pi^{\exists}$, $\exists w \in X$ *such that* $\alpha \in w$.

*Proof.* ($\Rightarrow$) Let $X_\alpha = \{w \mid \widehat{w} \wedge \alpha \text{ is consistent}\}$. Suppose $\widehat{u} \wedge \langle \pi \rangle^{\exists} \alpha$ is consistent. From axiom (A2a) we get $\exists w \in X_\alpha$ such that $\widehat{u} \wedge \langle \pi \rangle^{\exists} \widehat{w}$ is consistent. From Lemma 1.3, there exists $X \subseteq W$ such that $(u, X) \in R_\pi^{\exists}$ and $\vdash \widehat{w} \supset \widetilde{X}$. Since $\vdash \widehat{w} \supset \alpha$, we have $\exists (u, X) \in R_\pi^{\exists}$, $\exists w \in X$ such that $\alpha \in w$.
($\Leftarrow$) Suppose $\exists (u, X) \in R_\pi^{\exists}$, $\exists w \in X$ such that $\alpha \in w$. We need to show that $\widehat{u} \wedge \langle \pi \rangle^{\exists} \alpha$ is consistent. This is done by induction on the structure of $\pi$.

- The case when $\pi = t \in \mathcal{P}(\mathcal{V})$ follows from Lemma 1.4. For $\pi = \pi_1 \cup \pi_2$ the result follows from axiom (A7).
- $\pi = \pi_1; \pi_2$: Suppose $(u, X) \in R_{\pi_1;\pi_2}^{\exists}$ and $\exists w \in X$ such that $\alpha \in w$. From the definition of $R^{\exists}$ we get that there exists $Y \subseteq W$ such that $(u, Y) \in R_{\pi_1}^{\exists}$ and $\exists v \in Y$ such that $(v, X) \in R_{\pi_2}^{\exists}$. By induction hypothesis we have $\widehat{v} \wedge \langle \pi_2 \rangle^{\exists} \alpha$ is consistent. By definition of closure we have $\langle \pi_2 \rangle^{\exists} \alpha \in CL(\alpha_0)$. Therefore we get $\langle \pi_2 \rangle^{\exists} \alpha \in v$. Again applying induction hypothesis we get that $\widehat{u} \wedge \langle \pi_1 \rangle^{\exists} \langle \pi_2 \rangle^{\exists} \alpha$ is consistent. From (A8) we get $\widehat{u} \wedge \langle \pi_1; \pi_2 \rangle^{\exists} \alpha$ is consistent.
- $\pi = \pi_1^*$: From definition of $R^{\exists}$ there must be sets $Y_1, \ldots, Y_k$ such that $u \in Y_1$, $X = Y_k$ and for all $j : 1 < j < k$, $\exists v_j \in Y_j$ such that $(v_j, X_{j+1}) \in R_{\pi_1}^{\exists}$. From (A9) we get $\widehat{w} \wedge \langle \pi_1^* \rangle^{\exists} \alpha$ is consistent. By definition of closure, we have $\langle \pi_1 \rangle^{\exists} \langle \pi_1^* \rangle^{\exists} \alpha \in CL(\alpha_0)$. By induction hypothesis, $\widehat{v_{k-1}} \wedge \langle \pi_1 \rangle^{\exists} \langle \pi_1^* \rangle^{\exists} \alpha$ is consistent and therefore

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

from (A9) $\widehat{v}_{k-1} \wedge \langle \pi_1^* \rangle^{\exists} \alpha$ is consistent. Continuing in this manner we get $\widehat{u} \wedge \langle \pi_1^* \rangle^{\exists} \alpha$ is consistent.

□

LEMMA 1.6. *For all $t \in \mathcal{P}(\mathcal{V})$, for all $X \subseteq W$ and for all $u \in W$ the following holds:*

1. *if $(u, X) \in R_t^{\Box}$ then $\widehat{u} \wedge \langle t \rangle^{\Box} \widetilde{X}$ is consistent.*
2. *if $\widehat{u} \wedge \langle t \rangle^{\Box} \widetilde{X}$ is consistent then there exists $X' \subseteq X$ such that $(u, X') \in R_t^{\Box}$.*

*Proof.* The proof is by induction of the structure of the atomic tree $t$.

Let $t = (x)$.

Suppose $(u, X) \in R_t^{\Box}$ then from semantics we have $X = \{w \mid u \rightsquigarrow w\}$. This implies that for all $w \in X$, $\widehat{w} \wedge \widetilde{X}$ is consistent. From Lemma 1.1 we get $\widehat{u} \wedge \Box \widetilde{X}$ is consistent. From axiom (A5) case (C3) we get $\widehat{u} \wedge \langle t \rangle^{\Box} \widetilde{X}$ is consistent.

Suppose $\widehat{u} \wedge \langle t \rangle^{\Box} \widetilde{X}$ is consistent then by axiom (A5) case (C3) we have $\widehat{u} \wedge \Box \widetilde{X}$ is consistent. By Lemma 1.1 we get for all $w$ such that $u \rightsquigarrow w$, $\widehat{w} \wedge \widetilde{X}$ is consistent. Let $X' = \{w \mid u \rightsquigarrow w\}$, it is easy to see that $X' \subseteq X$ and thus from the semantics we get $(u, X') \in R_t^{\Box}$.

Let $t = (x, a_1, t_{a_1}) + \ldots + (x, a_k, t_{a_k})$.

Suppose $(u, X) \in R_t^{\Box}$. From semantics we have $\forall w_l \in \mathcal{I}(u), \forall a_j \in A, \forall v \in (w_l, a_j)_{\rightarrow}$ there exists $X_{l,j}^v$ such that $(v, X_{l,j}^v) \in R_{t_{a_j}}^{\Box}$. This implies that for all $w_l \in \mathcal{I}(u)$ and for all $a_j \in A$, $\widehat{w_l} \wedge \langle a_j \rangle \top$ is consistent and therefore $\widehat{u} \wedge \Box \langle a_j \rangle \top$ is consistent. By applying induction hypothesis and due to the fact that $\vdash \widetilde{X_{j,l}^v} \supset \widetilde{X}$ we get that $\widehat{v} \wedge \langle t_{a_j} \rangle^{\Box} \widetilde{X}$ is consistent for all $w_l \in \mathcal{I}(u)$, for all $a_j \in A$ and for all $v \in (w_l, a_j)_{\rightarrow}$. Thus from Lemma 1.1 and axiom (A5) case (C6) we can deduce that $\widehat{u} \wedge \langle t \rangle^{\Box} \widetilde{X}$ is consistent.

Suppose $\widehat{u} \wedge \langle t \rangle^{\Box} \widetilde{X}$ is consistent. From axiom (A5) case (C6) we get that for all $w_l \in \mathcal{I}(u)$, for all $a_j \in A$, $\widehat{w_l} \wedge \langle a_j \rangle \top$ is consistent. This implies that $(w_l, a_j)_{\rightarrow} \neq \emptyset$. From axiom (A5) case (C6) we also have that $\widehat{u} \wedge \Box [a_j] \langle t_{a_j} \rangle^{\Box} \widetilde{X}$ is consistent for all $a_j \in A$ and therefore for all $w_l \in \mathcal{I}(u)$, for all $a_j \in A$, for all $v \in (w_l, a_j)_{\rightarrow}$, $\widehat{v} \wedge \langle t_{a_j} \rangle^{\Box} \widetilde{X}$ is consistent. By induction hypothesis, there exists $X_{l,j}^v \subseteq X$ such that $(v, X_{l,j}^v) \in R_{t_{a_j}}^{\Box}$. Let $X' = \bigcup_{l=1,\ldots,m} \bigcup_{j=1,\ldots,k} \bigcup_{v \in (w_l, a_j)_{\rightarrow}} X_{l,j}^v$, by definition of $R_t^{\Box}$ we have $(u, X') \in R_t^{\Box}$. □

LEMMA 1.7. *For all $\pi \in \Gamma$, for all $X \subseteq W$ and $u \in W$, if $\widehat{u} \wedge \langle \pi \rangle^{\Box} \widetilde{X}$ is consistent then there exists $X' \subseteq X$ such that $(u, X') \in R_{\pi}^{\Box}$.*

*Proof.* By induction on the structure of $\pi$.

- $\pi = t \in \mathcal{P}(\mathcal{V})$: Suppose $\widehat{u} \wedge \langle t \rangle^{\Box} \widetilde{X}$ is consistent. From Lemma 1.6 item 2, it follows that there exists $X' \subseteq X$ such that $(u, X') \in R_{\pi}^{\Box}$.
- $\pi = \pi_1 \cup \pi_2$: By axiom (A7) we get $\widehat{u} \wedge \langle \pi_1 \rangle^{\Box} \widetilde{X}$ is consistent or $\widehat{u} \wedge \langle \pi_2 \rangle^{\Box} \widetilde{X}$ is consistent. By induction hypothesis there exists $X_1 \subseteq X$ such that $(u, X_1) \in R_{\pi_1}^{\Box}$ or there exists $X_2 \subseteq X$ such that $(u, X_2) \in R_{\pi_2}^{\Box}$. Hence we have $(u, X_1) \in R_{\pi_1 \cup \pi_2}^{\Box}$ or $(u, X_2) \in R_{\pi_1 \cup \pi_2}^{\Box}$.
- $\pi = \pi_1; \pi_2$: By axiom (A8), $\widehat{u} \wedge \langle \pi_1 \rangle^{\Box} \langle \pi_2 \rangle^{\Box} \widetilde{X}$ is consistent. Hence $\widehat{u} \wedge \langle \pi_1 \rangle^{\Box} (\bigvee (\widehat{w} \wedge \langle \pi_2 \rangle^{\Box} \widetilde{X}))$ is consistent, where the join is taken over all $w \in Y = \{w \mid w \wedge \langle \pi_2 \rangle^{\Box} \widetilde{X}$ is consistent$\}$. So $\widehat{u} \wedge \langle \pi_1 \rangle^{\Box} \widetilde{Y}$ is consistent. By induction hypothesis, there exists $Y' \subseteq Y$ such that $(u, Y') \in R_{\pi_1}^{\Box}$. We also have that for all $w \in Y$, $\widehat{w} \wedge \langle \pi_2 \rangle^{\Box} \widetilde{X}$ is consistent. Therefore we get for all $w_j \in Y' = \{w_1, \ldots, w_k\}$, $\widehat{w}_j \wedge \langle \pi_2 \rangle^{\Box} \widetilde{X}$ is consistent. By induction hypothesis, there exists $X_j \subseteq X$ such that $(w_j, X_j) \in R_{\pi_2}^{\Box}$. Let $X' = \bigcup_{j=1,\ldots,k} X_k \subseteq X$, we get $(u, X') \in R_{\pi_1;\pi_2}^{\Box}$.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

- $\pi = \pi_1^*$: Let $Z$ be the least set containing $X$ and closed under the condition: for all $w$, if $\widehat{w} \wedge \langle \pi_1 \rangle^{\Box} \widetilde{Z}$ is consistent, then $w \in Z$. By definition of $Z$ and induction hypothesis, we get for all $w \in Z$, there exists $X_w \subseteq X$ such that $(w, X_w) \in R^{\Box}_{\pi_1^*}$. It is also easy to see that $\vdash \widetilde{X} \supset \widetilde{Z}$. Using standard techniques, it is also easy to show that $\vdash \langle \pi_1 \rangle^{\Box} \widetilde{Z} \supset \widetilde{Z}$.
  Applying the induction rule ($IND_{\Box}$), we have $\vdash \langle \pi_1^* \rangle^{\Box} \widetilde{Z} \supset \widetilde{Z}$. By assumption, $\widehat{u} \wedge \langle \pi_1^* \rangle^{\Box} \widetilde{X}$ is consistent. So $\widehat{u} \wedge \langle \pi_1^* \rangle^{\Box} \widetilde{Z}$ is consistent. Hence $\widehat{u} \wedge \widetilde{Z}$ is consistent and therefore $u \in Z$. Thus we have $(u, X') \in R^{\Box}_{\pi_1^*}$ for some $X' \subseteq X$.

□

LEMMA 1.8. *For all $\langle \pi \rangle^{\Box} \alpha \in CL(\alpha_0)$, for all $u \in W$, $\widehat{u} \wedge \langle \pi \rangle^{\Box} \alpha$ is consistent iff there exists $(u, X) \in R^{\Box}_{\pi}$ such that $\forall w \in X$, $\alpha \in w$.*

*Proof.* ($\Rightarrow$) Follows from Lemma 1.7 (consider the set $X_\alpha = \{w \in W \mid \alpha \in w\}$).
($\Leftarrow$) Suppose $\exists (u, X) \in R^{\Box}_{\pi}$ such that $\forall w \in X$, $\alpha \in w$. We need to show that $\widehat{u} \wedge \langle \pi \rangle^{\Box} \alpha$ is consistent, this is done by induction on the structure of $\pi$.

- The case when $\pi = t \in \mathcal{P}(\mathcal{V})$ follows from Lemma 1.6. For $\pi = \pi_1 \cup \pi_2$ the result follows from axiom (A7).
- $\pi = \pi_1; \pi_2$: Since $(u, X) \in R^{\Box}_{\pi_1;\pi_2}$, there exists $Y = \{v_1, \ldots, v_k\}$, there exists sets $X_1, \ldots, X_k \subseteq X$ such that $\bigcup_{j=1,\ldots,k} X_j = X$, for all $j : 1 \leq j \leq k$, $(v_j, X_j) \in R_{\pi_2}$ and $(u, Y) \in R^{\Box}_{\pi_1}$. By induction hypothesis, for all $j$, $\widehat{v_j} \wedge \langle \pi_2 \rangle^{\Box} \alpha$ is consistent. Since $v_j$ is an atom and $\langle \pi_2 \rangle^{\Box} \alpha \in CL(\alpha_0)$, we get $\langle \pi_2 \rangle^{\Box} \alpha \in v_j$. Again by induction hypothesis we have $\widehat{u} \wedge \langle \pi_1 \rangle^{\Box} \langle \pi_2 \rangle^{\Box} \alpha$ is consistent. Hence from (A8) we have $\widehat{u} \wedge \langle \pi_1; \pi_2 \rangle^{\Box} \alpha$ is consistent.
- $\pi = \pi_1^*$: If $u \in X$ then $\vdash \widehat{u} \supset \widetilde{X}$. We have $\vdash \widetilde{X} \supset \alpha$ and hence we get $\widehat{u} \wedge \alpha$ is consistent. From axiom (A9) we have $\widehat{u} \wedge \langle \pi_1^* \rangle^{\Box} \alpha$ is consistent.
  Else we have $(u, X) \in R^{\Box}_{\pi_1;\pi_1^*}$. Let $Z_0 = X$ and $Z_{n+1} = Z_n \cup \{w \mid (w, Z') \in R^{\Box}_{\pi_1}, Z' \subseteq Z_n\}$. Take the least $m$ such that $u \in Z_m$. We have for all $w \in Z_{m-1}$, $\vdash \widehat{w} \supset \langle \pi_1^* \rangle^{\Box} \widetilde{X'}$ for some $X' \subseteq X$. We also have $(u, Z'_m) \in R^{\Box}_{\pi_1}$ for some $Z'_m = \{v_1, \ldots, v_k\} \subseteq Z_m$. Let $X_1, \ldots, X_k \subseteq X$ such that $\forall j : 1 \leq j \leq k$, we have $(v_j, X_j) \in R^{\Box}_{\pi_1^*}$ and $X' = \bigcup_{j=1,\ldots,k} X_j$. By an argument similar to the previous case we can show that $\widehat{u} \wedge \langle \pi_1 \rangle^{\Box} \langle \pi_1^* \rangle^{\Box} \widetilde{X'}$ is consistent. Hence we get $\widehat{u} \wedge \langle \pi_1; \pi_1^* \rangle^{\Box} \alpha$ is consistent. Therefore from axiom (A9) we have $\widehat{u} \wedge \langle \pi_1^* \rangle^{\Box} \alpha$ is consistent.

□

A routine induction gives us the following lemma from which Theorem 4.13 follows using the usual argument.

LEMMA 1.9. *For all $\beta \in CL(\alpha_0)$, for all $u \in W$, $M, u \models \beta$ iff $\beta \in u$.*

**B Proposition 6.3.** For technical convenience we assume that the uncertainty relation $\rightsquigarrow$ is reflexive. We can also extend the definition of the relation $R^{\mathcal{Q}}_{\pi}$ in the standard manner, where $R^{\exists}_{\pi}$ would represent a subjective path in the structure $(\mathcal{G}, u) \upharpoonright t$. In particular, for an "epsilon free" expression $\pi$, this would coincide with a deterministic (objective) path in $(\mathcal{G}, u) \upharpoonright t$. Consider the subclass of protocol trees $\mathcal{P}^{\Box}_{\epsilon}(\mathcal{V}) \subseteq \mathcal{P}_{\epsilon}(\mathcal{V})$ satisfying the following conditions: $t \in \mathcal{P}^{\Box}_{\epsilon}(\mathcal{V})$ iff

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

- For all maximal paths $\rho : s_1 z_1 \ldots z_{k-1} s_k \in \mathit{Paths}(T_t)$ we have
  - $z_1 = z_{k-1} = \epsilon$.
  - for all $j : 1 \leq j < k-1, z_{j+1} \in \Sigma$ if $z_j = \epsilon$ and $z_{j+1} = \epsilon$ if $z_j \in \Sigma$. That is, the labels on the path strictly alternate between $\epsilon$ and an action symbol in $\Sigma$.
- for all $s \in S_t$, if $\epsilon \in \mathcal{A}(s)$ then $\mathcal{A}(s) = \{\epsilon\}$. That is, the $\epsilon$ edge is never combined with $\Sigma$ in the branching structure.

We show that if the uncertainty relation is allowed to be specified in the protocol tree then the modalities $\langle \pi \rangle^{\Box}$ and $\langle \pi \rangle^{\Diamond}$ can be eliminated. Formally, let $\mathcal{L}'_{EPL}$ be the fragment of the language $\mathcal{L}_{EPL}$ (defined in Section §4) which does not include formulas of the form $\langle \pi \rangle^{\Box}\alpha$ and $\langle \pi \rangle^{\Diamond}\alpha$. We can show the following translation result.

PROPOSITION 2.1. *For all $\alpha \in \mathcal{L}_{EPL}$, there exists $\alpha' \in \mathcal{L}'_{EPL}$ such that $M, u \models \alpha$ iff $M, u \models \alpha'$.*

*Proof.* We present a proof sketch here. The idea is to translate the constructs $\langle \pi \rangle^{\Box}$ and $\langle \pi \rangle^{\Diamond}$ into $\langle \pi' \rangle^{\forall}$ and $\langle \pi' \rangle^{\exists}$, respectively, where $\pi'$ is a composite tree expression over the expanded set $\mathcal{P}^{\Box}_{\epsilon}(\mathcal{V})$.

The interesting case is when $\pi$ is atomic, that is, $\pi = t \in \mathcal{P}(\mathcal{V})$. We define a translation function $[\![\cdot]\!] : \mathcal{P}(\mathcal{V}) \to \mathcal{P}^{\Box}_{\epsilon}(\mathcal{V})$ inductively as follows:

- if $t = (x)$, $[\![t]\!] = (y_1, \epsilon, (x, \epsilon, y_2))$. In other words, the single node tree is expanded to a path which is prefixed and suffixed with an $\epsilon$ edge.
- if $t = (x, a_1, t_{a_1}) + \ldots + (x, a_k, t_{a_k})$ we define $[\![t]\!] = (y, \epsilon, t_x)$ where $t_x = (x, a_1, [\![t_{a_1}]\!]) + \ldots + (x, a_k, [\![t_{a_k}]\!])$.

The translation function $[\![\cdot]\!]$ can be extended to the compositional operators as well as to formulas in the obvious manner where we have,

- $[\![\langle \pi \rangle^{\Box}\alpha]\!] = \langle [\![\pi]\!] \rangle^{\forall} [\![\alpha]\!]$.
- $[\![\langle \pi \rangle^{\Diamond}\alpha]\!] = \langle [\![\pi]\!] \rangle^{\exists} [\![\alpha]\!]$.

It is then an easy inductive argument to show that the translation preserves the satisfaction relation. In other words, $M, u \models \alpha$ iff $M, u \models [\![\alpha]\!]$.

## BIBLIOGRAPHY

Balbiani, P., Herzig, A., & Troquard, N. (2007). Alternative axiomatics and complexity of deliberative STIT theories. *Journal of Philosophical Logic*, **37**(4), 387–406.

Blackburn, P., de Rijke, M., & Venema, Y. (2002). *Modal Logic*. Cambridge: Cambridge University Press.

Bratman, M. (1987). *Intention, Plans and Practical Reason*. Cambridge: Harvard University Press.

Bratman, M. (1999). *Faces of Intention*. Cambridge: Cambridge University Press.

Broersen, J. (2008). A logical analysis of the interaction between 'obligation-to-do' and 'knowingly doing'. In van der Torre, L., and van der Meyden, R., editors. *Proceedings 9th International Workshop on Deontic Logic in Computer Science (DEON'08)*, Vol. 5076 of *LNCS*. Berlin: Springer, pp. 140–154.

Carr, D. (1979). The logic of knowing how and ability. *Mind*, **88**, 394–409.

Cohen, P. R., & Levesque, H. (1990). Intention is choice with committment. *Artificial Intelligence*, **42**(3), 213–261.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

Cross, C. B. (1986). 'Can' and the logic of ability. *Philosophical Studies*, **50**(1), 53–64.
Elgesem, D. (1997). The modal logic of agency. *Nordic Journal of Philosophical Logic*, **2**(2), 1–46.
Fagin, R., Halpern, J., Moses, Y., & Vardi, M. (1995). *Reasoning about Knowledge*. Boston: The MIT Press.
Fagin, R., Halpern, J., Moses, Y., & Vardi, M. (1997). Knowledge-based programs. *Distributed Computing*, **10**(4), 199–225.
Fantl, J. (2008). Knowing-how and knowing-that. *Philosophy Compass*, **3**(3), 451–470.
Gilbert, M. (1989). *On Social Facts*. Princeton: Princeton University Press.
Goranko, V., & Jamroga, W. (2004). Comparing semantics of logics for multi-agent systems. *Synthese: Knowledge, Rationality, and Action*, **139**(2), 241–280.
Governatori, G., & Rotolo, A. (2005). On the axiomatization of elgesem's logic of agency and ability. *Journal of Philosophical Logic*, **34**(4), 403–431.
Halpern, J., & Fagin, R. (1989). Modelling knowledge and action in distributed systems. *Distributed Computing*, **3**(4), 159–177.
Halpern, J., & Moses, Y. (1990). Knowledge and common knowledge in a distributed environment. *Journal of the ACM*, **37**(3), 549–587.
Halpern, J. Y., van der Meyden, R., & Vardi, M. Y. (2004). Complete axiomatizations for reasoning about knowledge and time. *SIAM Journal on Computing*, **33**(3), 674–703.
Harel, D., Kozen, D., & Tiuryn, J. (2000). *Dynamic Logic*. Boston: The MIT Press.
Herzig, A., Long, J., Longin, D., & Polacsek, T. (2000). A logic for planning under partial observability. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence*. Austin, Texas: AAAI Press, pp. 768–773.
Herzig, A., & Troquard, N. (2006). Knowing how to play: Uniform choices in logics of agency. In *AAMAS '06: Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems*. Hakodate, Japan: ACM, pp. 209–216.
Horty, J. (2001). *Agency and Deontic Logic*. Oxford: Oxford University Press.
Hoshi, T. (2009). Epistemic dynamics and protocol information. PhD Thesis, Stanford University.
Icard, T., Pacuit, E., & Shoham, Y. (2010). Joint revision of beliefs and intentions. In *Proceedings of KR 2010*, 572–574.
Jamroga, W., & Agotnes, T. (2007). Constructive knowledge: What agents can achieve under imperfect information. *Journal of Applied Non-Classical Logics*, **17**(4), 423–475.
Kozen, D., & Parikh, R. (1981). An elementary proof of the completeness of PDL. *Theoretical Computer Science*, **14**, 113–118.
Lesperance, Y., Levesque, H., Lin, F., & Scherl, R. (2000). Ability and knowing how in the situation calculus. *Studia Logica*, **65**, 165–186.
Lorini, E., & Herzig, A. (2008). A logic of intention and attempt. *Synthese*, **163**(1), 45–77.
Lorini, E., Schwarzentruber, F., & Herzig, A. (2009). Epistemic games in modal logic: joint actions, knowledge and preferences all together. In *Proceedings of LORI'09*. Berlin: Springer-Verlag, pp. 212–226.
Meyer, J.-J., van der Hoek, W., & van Linder, B. (1999). A logical approach to the dynamics of commitments. *Artificial Intelligence*, **113**, 1–40.
Meyer, J.-J., & Veltman, F. (2007). Intelligent agents and common sense reasoning. In Blackburn, P., van Benthem, J., and Wolter, F., editors. *Handbook of Modal Logic*. Amsterdam: Elsevier, pp. 991–1029.
Mukund, M., & Sohoni, M. (1997). Keeping track of the latest gossip in a distributed system. *Distributed Computing*, **10**(3), 137–148.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

Nicola, R. D. (1987). Extensional equivalences for transition systems. *Acta Informatica*, **24**, 211–237.

Parikh, R. (1985). The logic of games and its applications. In *Topics in the Theory of Computation (Borgholm, 1983)*, Vol. 102 of *North-Holland Math. Stud.* Annals of Discrete Mathematics, **24** (1985) 111–140.

Parikh, R., & Ramanujam, R. (2003). A knowledge based semantics of messages. *Journal of Logic, Language and Information*, **12**, 453–467.

Peleg, D. (1987). Concurrent dynamic logic. *Journal of the ACM*, **34**(2), 450–479.

Ramanujam, R. (1996). Local knowledge assertions in a changing world. In *Proceedings of TARK*. San Francisco: Morgan Kaufmann, pp. 1–17.

Ramanujam, R., & Simon, S. (2009). Dynamic logic of tree composition. In *Perspectives in Concurrency Theory*. India: Universities Press, pp. 408–430.

Roy, O., & Pacuit, E. (2011). Substantive assumptions in interaction: A logical perspective. Manuscript.

Ryle, G. (1949). *The Concept of Mind*. Chicago: The University of Chicago Press.

Schmidt, R., & Tishkovsky, D. (2008). On combinations of propositional dynamic logic and doxastic modal logics. *Journal of Logic, Language and Information*, **17**, 109–129.

Searle, J. (1990). Collective intentions and actions. In Cohen, P., Morgan, J., and Pollack, M., editors. *Intentions in Communication*. Boston: The MIT Press, pp. 401–415.

Singh, M. P. (1999). Know-how. In Wooldridge, M., and Rao, A., editors. *Foundations of Rational Agency*. Berlin: Springer, pp. 105–132.

Stanley, J., & Williamson, T. (2001). Knowing how. *Journal of Philosophy*, **98**, 411–444.

Sugden, R. (2003). The logic of team reasoning. *Philosophical Explorations*, **6**(3), 165–181.

Tuomela, R. (2010). *The Philosophy of Sociality*. Oxford: Oxford University Press.

van Benthem, J. (2001a). Extensive games as process models. *Journal of Logic, Language and Information*, **11**, 289–313.

van Benthem, J. (2001b). Games in dynamic epistemic logic. *Bulletin of Economic Research*, **53**(4), 219–248.

van Benthem, J. (2008). *In Praise of Strategies*. Technical report, ILLC Technical Reports.

van Benthem, J. (2011). *Logical Dynamics of Information and Interaction*. Cambridge: Cambridge University Press.

van Benthem, J., Gerbrandy, J., Hoshi, T., & Pacuit, E. (2009). Merging frameworks for interaction. *Journal of Philosophical Logic*, **38**(5), 491–526.

van Benthem, J., & Pacuit, E. (2006). The tree of knowledge in action: Towards a common perspective. In Governatori, G., Hodkinson, I., and Venema, Y., editors. *Proceedings of Advances in Modal Logic*, Vol. 6. London: King's College Press, pp. 87–106.

van der Hoek, W., van Linder, B., & Meyer, J.-J. (1998). Formalising abilities and opportunities of agents. *Fundamenta Informaticae*, **34**, 1–49.

van der Hoek, W., & Wooldridge, M. (2003a). Cooperation, knowledge, and time: Alternating-time temporal epistemic logic and its applications. *Studia Logica*, **75**, 125–157. doi:10.1023/A:1026185103185.

van der Hoek, W., & Wooldridge, M. (2003b). Towards a logic of rational agency. *Logic Journal of the IGPL*, **11**(2), 135–160.

van der Meyden, R. (1996). Finite state implementations of knowledge-based programs. In *Proceedings of the Conference on Foundations of Software Technology and Theoretical Computer Science*, Vol. 1180 of *Lecture Notes in Computer Science*. Berlin: Springer, pp. 262–273.

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press

van Eijk, J., Kuppusamy, L., & Wang, Y. (2009). Verifying epistemic protocols under common knowledge. In Heifetz, A., editor. *Proceedings of TARK*. Stanford: ACM, pp. 257–266.
van Otterloo, S. (2005). A strategic analysis of multi-agent protocols. PhD Thesis, ILLC University of Amsterdam.
Wang, Y. (2010). Epistemic modelling and protocol dynamics. PhD Thesis, CWI.
Williams, J. (2008). Propositional knowledge and know-how. *Synthese*, **165**, 107–125.
Zielonka, W. (1987). Notes on finite asynchronous automata. *RAIRO Informatique Théorique et Applications*, **21**(2), 99–135.

□

TILBURG INSTITUTE FOR LOGIC AND PHILOSOPHY OF SCIENCE
TILBURG UNIVERSITY
TILBURG, 5000LE, THE NETHERLANDS
*E-mail*: e.j.pacuit@uvt.nl

CENTRUM FOR WISKUNDE & INFORMATICA
SCIENCE PARK 123
AMSTERDAM, 1098XG, THE NETHERLANDS
*E-mail*: s.e.simon@cwi.nl

https://doi.org/10.1017/S1755020311000190 Published online by Cambridge University Press