Hidden Protocols:
Modifying our expectations in an evolving world

Hans van Ditmarsch[a], Sujata Ghosh[b], Rineke Verbrugge[c], Yanjing Wang[d,*]

[a]*LORIA, CNRS – Université de Lorraine, France*
[b]*Indian Statistical Institute, Chennai, India*
[c]*Institute of Artificial Intelligence, University of Groningen, The Netherlands*
[d]*Department of Philosophy, Peking University, China*

Abstract

When agents know a protocol, this leads them to have expectations about future observations. Agents can update their knowledge by matching their actual observations with the expected ones. They eliminate states where they do not match. In this paper, we study how agents perceive protocols that are not commonly known, and propose a semantics-driven logical framework to reason about knowledge in such scenarios.

In particular, we introduce the notion of epistemic expectation models and a propositional dynamic logic-style epistemic logic for reasoning about knowledge via matching agents' expectations to their observations. It is shown how epistemic expectation models can be obtained from epistemic protocols. Furthermore, a characterization is presented of the effective equivalence of epistemic protocols. We introduce a new logic that incorporates updates of protocols and that can model reasoning about knowledge and observations. Finally, the framework is extended to incorporate fact-changing actions, and a worked-out example is given.

Keywords: protocols, dynamic epistemic logic, guarded automata

*Corresponding author

Email addresses: hans.van-ditmarsch@loria.fr (Hans van Ditmarsch), sujata@isichennai.res.in (Sujata Ghosh), rineke@ai.rug.nl (Rineke Verbrugge), y.wang@pku.edu.cn (Yanjing Wang)

1. Introduction

Talking about knowledge and protocols, some questions come to our minds: *What do we mean by knowing a protocol? How does this protocol knowledge affect our knowledge of facts about the world?* The literature abounds with various formal models answering these questions from different angles [1, 2, 3, 4, 5], and the proper representation and formalization of knowledge and knowledge dynamics is a core interest in the area of artificial intelligence [6, 7, 8, 9]. In some situations, agents have partial knowledge of the underlying protocols that guide the behaviors of other agents. Based on their incomplete knowledge of protocols and their observations, the agents try to reason about other agents' epistemic attitudes as well as about hard facts. Protocols play a role, for example, when agents communicate using full-blown secret codes (see [10] for many intriguing historical examples). Our daily communications provide more mundane protocols that may help to hide information from part of the participants.

Example 1 (The voice of Kathleen Ferrier). *Consider a café in the 1950s, with three persons, Kate, Jane and Ann sitting across a table. Suppose Kate is gay and wants to know whether either of the other two is gay. She wants to convey the right information to the right person, without the other getting any idea of the information that is being communicated. She states, 'I am musical, I like Kathleen Ferrier's voice'. Jane, who is gay herself, immediately realizes that Kate is gay, whereas, for Ann, the statement just conveys a particular taste in music.*[1]

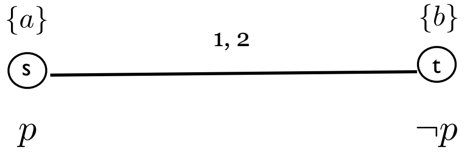
Example 2 (Valentine's Day). *Coming back to the present day, consider a similar café scenario with Carl, Ben and Alice. Carl and Ben are childhood friends and know each other like the back of their hands. Carl says to Ben: 'On Valentine's day I went to the pub with Mike and Sara. It was a crazy night!' This immediately catches the attention of Alice, who is in love with Mike. She asks: 'What happened?' Carl winks to Ben and says: 'Nothing'. Knowing Carl very well, Ben immediately realizes that indeed nothing has*

[1] This example has been inspired by the interviews in [11], from which it appears that in 1950s Amsterdam, 'musical' was indeed a code term for 'gay', known almost exclusively by gay people. The additional mention of singer Kathleen Ferrier strengthened this 'gay' hint. Among gay women, Ferrier's low contralto voice, for example in her performance as Orfeo in Gluck's Orfeo ed Euridice, was widely popular.

happened, whereas Alice becomes unsure of that, as she saw the wink that Carl has given to Ben.

This paper presents a dynamic epistemic logic (DEL, [12, 13]) that can suitably describe such scenarios. Knowing a protocol can mean 'knowing what to do according to the protocol' [1]. It can also correspond to 'understanding the underlying meaning of the actions induced by the protocol' [2]. Here, we follow the latter interpretation, which appears to capture the notion of a protocol in the types of situations we want to model. Kate's making a statement like 'I am musical, I like Kathleen Ferrier's voice' corresponds to the fact that 'Kate is gay'. In the second situation, 'Nothing' (even if accompanied by a wink) corresponds to the fact that 'Nothing has happened'.

Our work is inspired by two lines of research: the work relating dynamic epistemic logic (DEL) and epistemic temporal logic (ETL) [3, 5, 14] and the work on protocol changes [4, 15]. In [14], Pacuit and Simon model protocols as tree compositions, basically equating protocols with plans. Hoshi *et al.* [3, 5] propose the notion of 'state-dependent' DEL-protocols (sets of sequences of *event models* [13]) in order to handle protocols that are not common knowledge. Consider an epistemic scenario wherein the agents are not only uncertain about the factual state of the world but also about the protocol that can be executed given some factual state, depicted as the model:



In this model, s, t are possible worlds, p is a proposition, and a, b are expected actions. The uncertainty of the agents about the protocol is denoted by a state-dependent protocol assigning singleton action sets $\{a\}$ to s and $\{b\}$ to t. Note that we have omitted the reflexive arrows for agents 1 and 2 for the sake of compact representation, and we will follow this convention throughout this paper. A system wherein the protocol can be different in any state is clearly more complex than a system wherein the protocol is a background parameter, and thus can be assumed common knowledge to all agents. But in the example model above, we can still reclaim some form of common knowledge of the protocol, namely by describing it intuitively as follows: **if** p **then** a and **if** $\neg p$ **then** b. In order to discuss the knowledge of protocols formally, we need to first fix a protocol specification language, which will then enable us to represent such protocol models in a more informative way.

Given a protocol language, how do we obtain such epistemic models with protocol information from specifications of conditional protocols, and vice versa? Similar questions are addressed in [4, 15], in which Wang presents a logical framework that incorporates protocol specifications in epistemic models and introduces the idea of matching observations to expectations. However, there, protocols are assumed to be common knowledge. We do not assume that here.

Our work is based on the logic developed in [4] but in the current article we use epistemic models with procedural information as in [3, 5] to deal with uncertainties about protocols, an agent's knowledge of underlying protocols, and her current observations affecting factual uncertainty. In our framework, the protocols can be viewed as 'given by nature', so the framework does not cover interesting aspects such as how and by whom the protocols have been designed and how agents have come to agree to use them.

The ingredients of our work are:

1. epistemic models encoding state-dependent expected observations;
2. an update mechanism for eliminating impossible worlds according to the observation of agents and their expectations;
3. a formal language for specifying observations and protocols;
4. protocol models that represent agents' incomplete information about the 'real' protocols;
5. an update mechanism for incorporating protocol information (as protocol models) in epistemic (observation) models;
6. a notion of equivalence between protocol models;
7. a logic for reasoning about knowledge based on protocols;
8. fact-changing actions and factual change systems, in order to investigate how we modify our expectations in an evolving world.

The paper is organized as follows. Section 2 introduces epistemic expectation models and a simple propositional dynamic logic (PDL)-style epistemic logic for reasoning about knowledge via matching agents' expectations to their observations. Section 3 discusses how we obtain epistemic expectation models from protocol models (i.e., epistemic protocols). We characterize three classes of epistemic expectation models that can be generated from various epistemic models. Furthermore we give a characterization of the effective equivalence of epistemic protocols. A logic is then given to incorporate the updates of protocols and to model reasoning about knowledge and observations. In Section 4 we address incorporation of fact-changing actions. Section 5 discusses the application of the full framework, including factual changes, to a well-known logic puzzle. Finally, we point out relations to other research and future work in Sections 6 and 7.

This article is the extended version of [16]. The main differences are: the introduction of the concept of *observational saturation* and a theorem about its relation to protocol models (Theorem 29); results about systems with fact-changing actions (Section 4); an extended application, namely about a protocol in the 'One hundred prisoners and a lightbulb' puzzle (Section 5); and a more extensive discussion of related work and ideas for future research (Sections 6 and 7).

2. Reasoning via Expectation and Observation

In this section, we introduce *epistemic expectation models*, which are Kripke models with expected observations. We propose a dynamic logic style epistemic logic that is interpreted on such models for reasoning about knowledge via matching observations with expectations.

2.1. Epistemic Expectation Models

Let $\mathbf{I}$ be a finite set of agents, and let $\mathbf{P}$ be a finite set of propositions describing the facts about the world. Let $Bool(\mathbf{P})$ denote the set of all Boolean formulas over $\mathbf{P}$. To set up the semantics, we first define a Kripke model in the usual sense, which models agents' epistemic uncertainties regarding the actual state of the world.

Definition 3 (Epistemic model). *An* epistemic model $\mathcal{M}_e$ *is a triple* $\langle S, \sim, V\rangle$*, where* S *is a non-empty domain of states,* $\sim$ *stands for a set of accessibility (equivalence) relations* $\{\sim_i \mid i \in \mathbf{I}\}$*, and* $V : S \to \mathcal{P}(\mathbf{P})$ *is a valuation assigning to each state a set of propositional variables (those that are 'true in that state').*

We will introduce the concept of *epistemic expectation models* based on Kripke models, which captures the expected observations of agents. Agents observe what is happening around them and reason based on these observations. Examples of such observations are 'making an announcement', 'going to the right', and 'nodding your head'. One can distinguish such observations of *actions* from observations of *facts*, such as 'the chair is red'. Factual observations are not ruled out in our framework but we typically have observations of actions in mind. To this end, we introduce a finite set of actions, named Σ. An *observation* is a finite string of actions, for example, $abcd$. Note that an agent may expect different (even infinitely many) potential observations to happen at a given state, for example, she may expect $a \dots ab$ to happen for any finite sequence of as preceding the terminating action b. As human beings and computers are essentially finite, we need to denote such expectations in a finitary way. To this end, we introduce the *observation expressions* (as regular expressions over Σ):

Definition 4 (Observation expressions). *Given a finite set of action symbols Σ, the language $\mathcal{L}_{obs}$ of* observation expressions *is defined by the following BNF:*

$$\pi \quad ::= \quad \delta \mid \ \varepsilon \mid a \mid \pi \cdot \pi \mid \pi + \pi \mid \pi^*$$

where δ stands for the empty set $\emptyset$ of observations, the constant ε represents the empty string, and $a \in \Sigma$.

The semantics for the observation expressions are given by *sets of observations* (strings over Σ), similar to those for regular expressions.

Definition 5 (Observations). *Given an observation expression π, the corresponding* set of observations*, denoted by $\mathcal{L}(\pi)$, is the set of finite strings over Σ defined as follows:*

$$\begin{aligned}
&\mathcal{L}(\delta) = \emptyset\\
&\mathcal{L}(\varepsilon) = \{\epsilon\}\\
&\mathcal{L}(a) = \{a\}\\
&\mathcal{L}(\pi \cdot \pi') = \{wv \mid w \in \mathcal{L}(\pi) \textit{ and } v \in \mathcal{L}(\pi')\}\\
&\mathcal{L}(\pi + \pi') = \mathcal{L}(\pi) \cup \mathcal{L}(\pi')\\
&\mathcal{L}(\pi^*) = \{\epsilon\} \cup \textstyle\bigcup_{n>0}(\mathcal{L}(\underbrace{\pi \cdots \pi}_{n}))
\end{aligned}$$

Now we are ready to define *epistemic observation models*, which can be seen as epistemic models together with, for each world, a set of potential or expected observations.

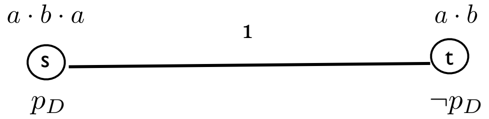
Definition 6 (Epistemic expectation model). *An* epistemic expectation model *$\mathcal{M}_{exp}$ is a quadruple*

$$\langle S, \sim, V, \mathit{Exp}\rangle,$$

where $\langle S, \sim, V\rangle$ is an epistemic model (the epistemic skeleton *of $\mathcal{M}_{exp}$) and $\mathit{Exp} : S \to \mathcal{L}_{obs}$ is an expected observation function assigning to each state an observation expression π such that $\mathcal{L}(\pi) \neq \emptyset$ (non-empty set of finite sequences of observations). An* epistemic expectation state *is a pointed epistemic expectation model $\langle S, \sim, V, \mathit{Exp}, s\rangle$. Intuitively, Exp assigns to each state a set of potential or expected observations.*

Given an epistemic expectation model $\mathcal{M}_{exp} = \langle S, \sim, V, \mathit{Exp}\rangle$, note that $\langle S, \sim, V\rangle$ is an epistemic model in the usual sense. Hence, sometimes, we also denote an epistemic expectation model as $(\mathcal{M}_e, \mathit{Exp})$, where $\mathcal{M}_e$ is the corresponding epistemic model. An epistemic model $\mathcal{M}_e$ can be considered as an epistemic expectation model $\mathcal{M}_{exp}$ where for all $s \in S$, $\mathit{Exp}(s) = \mathbf{\Sigma}^*$ (where $\mathbf{\Sigma}^*$ is shorthand for $(a_0 + a_1 + \cdots + a_k)^*$, given that $\mathbf{\Sigma} = \{a_0, \ldots, a_k\}$). Thus, in an epistemic model, the observations possible at each state are not specified; one can expect to observe anything. In this sense, $\mathcal{M}_e$ lacks in providing procedural information about the world, and $\mathcal{M}_{exp}$ fills that gap. In what follows we often leave out the subscripts, whenever the respective models are clear from the context.

Example 7 (Dutch or not Dutch). *In the Netherlands, people often greet each other by kissing three times on the cheek (left-right-left) while in the rest of Europe, people usually kiss each other only twice. We can reason whether a person is 'Dutch-related' by observing his behavior. Let p_D be the proposition meaning 'Simon is Dutch-related'; a and b are two actions denoting kissing the left cheek and kissing the right cheek, respectively. The following model is what we expect (reflexive arrows are omitted again):*



The indistinguishability relation above depicts that agent 1 does not know whether p_D. The associated observations are those that the agents might expect in each state. Intuitively, if agent 1 observes Simon kissing three times (observation aba), then he or she can infer that Simon is Dutch-related. In the next subsection, a simple logic is defined to handle such reasoning based on actual observations.

2.2. Public Observation Logic

In this subsection we define a simple dynamic logic with knowledge operators to reason about knowledge via the matching of observations and expectations. The idea is similar to the one behind public announcement logic, where people update their information by deleting impossible scenarios according to what is publicly announced. Here we relax the link between meaning and public actions (like an announcement). We assume that when observing an action, people delete some impossible scenarios where they wouldn't expect that observation to happen. To make such reasoning formal, we first define the update of epistemic expectation models according to some observation $w \in \mathbf{\Sigma}^*$. The idea behind an updated expectation model is that we delete the states where the observation w could not have been happened.

Definition 8 (Update by observation). *Let w be an observation over $\mathbf{\Sigma}$ and let $\mathcal{M} = (S, \sim, V, \mathit{Exp})$ be an epistemic expectation model. The updated model $\mathcal{M}|_w = (S', \sim', V', \mathit{Exp}')$. Here, $S' = \{s \mid \mathcal{L}(\mathit{Exp}(s)\backslash w) \neq \emptyset\}$, $\sim'_i = \sim_i|_{S' \times \mathbf{I} \times S'}$, $V' = V|_{S'}$, and $\mathit{Exp}'(s) = \mathit{Exp}(s)\backslash w$, where $\pi\backslash w$ is defined as the regular expression denoting the set $\{v \mid wv \in \mathcal{L}(\pi)\}$ ($\pi\backslash w$ corresponds to right residuation with respect to the monoid $(\mathbf{\Sigma}^*, \cdot, \varepsilon)$).*

A regular expression $\pi\backslash w$ is defined with an auxiliary output function o from the set of regular expressions over $\mathbf{\Sigma}$ to $\{\delta, \varepsilon\}$. If $\varepsilon \in \mathcal{L}(\pi)$, the output function o maps a regular expression π to ε; otherwise, it maps π to δ [17, 18]:

$$\begin{array}{ll}
\pi = o(\pi) + \sum_{a \in \mathbf{\Sigma}}(a \cdot \pi\backslash a) & \varepsilon\backslash a = \delta\backslash a = b\backslash a = \delta \quad (a \neq b) \\
o(\varepsilon) = \varepsilon & a\backslash a = \varepsilon \\
o(\delta) = o(a) = \delta & (\pi + \pi')\backslash a = \pi\backslash a + \pi'\backslash a \\
o(\pi + \pi') = o(\pi) + o(\pi') & (\pi \cdot \pi')\backslash a = (\pi\backslash a) \cdot \pi' + o(\pi) \cdot (\pi'\backslash a) \\
o(\pi \cdot \pi) = o(\pi) \cdot o(\pi') & \pi^*\backslash a = \pi\backslash a \cdot \pi^* \\
o(\pi^*) = \varepsilon & \pi\backslash a_0 \cdots a_n = \pi\backslash a_0\backslash a_1 \ldots \backslash a_n
\end{array}$$

The above construction of the output function helps to compute the residual of compositions. Reading from left to right the above equations can be viewed as rewriting rules which push the $\backslash a$ operation to the 'inner' part of the expression and finally eliminate them. Thus by using these equations we can compute residuals of observations syntactically.

We design the *Public observation logic* (POL) to reason about observations:

Definition 9 (Public observation logic). *The* formulas φ *of* POL *are given by:*

$$\varphi \quad ::= \quad \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid K_i\varphi \mid [\pi]\varphi$$

where $p \in \mathbf{P}$, $i \in \mathbf{I}$, and $\pi \in \mathcal{L}_{obs}$. The other propositional connectives are defined in the usual manner.

Intuitively, $[\pi]\varphi$ says that 'after any observation in π, φ holds'.

Definition 10 (Truth definition for POL**).** *Given an epistemic expectation model $\mathcal{M} = (S, \sim, V, Exp)$, a state $s \in S$, and a* POL*-formula φ, the truth of φ at s, denoted by $\mathcal{M}, s \vDash \varphi$, is defined as follows:*

$$\begin{array}{rcl}
\mathcal{M}, s \vDash p & \Leftrightarrow & p \in V(s) \\
\mathcal{M}, s \vDash \neg\varphi & \Leftrightarrow & \mathcal{M}, s \nvDash \varphi \\
\mathcal{M}, s \vDash \varphi \wedge \psi & \Leftrightarrow & \mathcal{M}, s \vDash \varphi \textit{ and } \mathcal{M}, s \vDash \psi \\
\mathcal{M}, s \vDash K_i\varphi & \Leftrightarrow & \textit{for all } t : (s \sim_i t \textit{ implies } \mathcal{M}, t \vDash \varphi) \\
\mathcal{M}, s \vDash [\pi]\varphi & \Leftrightarrow & \textit{for each } w \in \mathcal{L}(\pi) : (w \in \textit{init}(\textit{Exp}(s)) \textit{ implies } \mathcal{M}|_w, s \vDash \varphi)
\end{array}$$

where $w \in init(\pi)$ iff $\exists v \in \mathbf{\Sigma}^*$ such that $wv \in \mathcal{L}(\pi)$ (namely $\mathcal{L}(\pi\backslash w) \neq \emptyset$).

Consider the model $\mathcal{M}$ in Example 7. If we observe one or two kisses, first on the left and then on the right cheek ($a \cdot b$), agent 1 still cannot tell that Simon is Dutch-related ($\neg K_1 p_D$), but if there is one more kiss on the left cheek to follow (a), then agent 1 knows. Formally, it can be verified that $\mathcal{M}, s \vDash [a \cdot b](\neg K_1 p_D \wedge [a]K_1 p_D)$ (cf. Example 7). More complicated observation expressions π can be used to express (infinite) sets of observations, for example, $[\mathbf{\Sigma}^* \cdot a \cdot \mathbf{\Sigma}^*]K_i\varphi$ says 'as long as a is observed at some point, i knows φ' (recall that $\mathbf{\Sigma}^*$ denotes the expression corresponding to the set all observations).

Clearly, the standard bisimulation between epistemic models is not an invariance of the above logic: POL can reason about what may happen at each state. We now define bisimulation between epistemic expectation models, which facilitates characterization results in later sections.

Definition 11 (Observation bisimulation). *A binary relation R between the domains of two epistemic expectation models $\mathcal{M} = (S, \sim, V, \mathit{Exp})$ and $\mathcal{N} = (S', \sim', V', \mathit{Exp}')$ is called a* bisimulation *if for any $s \in S, s' \in S'$, we have that if $(s, s') \in R$, then the following conditions hold:*

Propositional invariance $V(s) = V'(s')$*;*

Observational invariance $\mathcal{L}(\mathit{Exp}(s)) = \mathcal{L}(\mathit{Exp}'(s'))$*;*

Zig *if $s \sim_i t$ in $\mathcal{M}$ then there exists a t' in $\mathcal{N}$ such that $s' \sim'_i t'$ and tRt';*

Zag *if $s' \sim'_i t'$ in $\mathcal{N}$ then there exists a t in $\mathcal{M}$ such that $s \sim_i t$ and tRt'.*

A bisimulation R is total *if every state in one model is linked by R to some state in the other model. $\mathcal{M}$ and $\mathcal{N}$ are said to be (total)* bisimilar *($\mathcal{M} \leftrightarroweq_o \mathcal{N}$) if there is a (total) bisimulation R between $\mathcal{M}$ and $\mathcal{N}$. $(\mathcal{M}, s)$ and $(\mathcal{N}, s')$ are said to be bisimilar ($\mathcal{M}, s \leftrightarroweq_o \mathcal{N}, s'$) if there is a bisimulation R between them such that $(s, s') \in R$.*

Note that the standard bisimilarity (notation $\leftrightarroweq$) is defined as $\leftrightarroweq_o$ without the condition for the invariance for observations. It is not hard to show that $\leftrightarroweq_o$ and logical equivalence $\equiv_{\text{POL}}$ coincide on finite models:

Proposition 12 (Bisimulation invariance). *For any two finite epistemic expectation states $\mathcal{M}, s$ and $\mathcal{N}, s'$, the following statements are equivalent:*

- $\mathcal{M}, s \leftrightarroweq_o \mathcal{N}, s'$
- *For any formula* $\varphi \in \text{POL} : \mathcal{M}, s \vDash \varphi \iff \mathcal{N}, s' \vDash \varphi$

Proof. $[\leftrightarroweq_o \implies \equiv_{\text{POL}}]$: We prove this by induction on φ. The Boolean and $K_i\psi$ cases are trivial. Now consider $\varphi = [\pi]\psi$; so suppose that $\mathcal{M}, s \leftrightarroweq_o \mathcal{N}, s'$ but $\mathcal{M}, s \vDash [\pi]\psi$ and $\mathcal{N}, s' \nvDash [\pi]\psi$. Then there exists a $w \in \mathcal{L}(\pi)$ such that $w \in \mathit{init}(\mathit{Exp}(s'))$ and $\mathcal{N}|_w, s' \vDash \neg\psi$.

By the definition of $\leftrightarroweq_o$, we have $\mathcal{L}(\mathit{Exp}(s)) = \mathcal{L}(\mathit{Exp}(s'))$, therefore $w \in \mathit{init}(\mathit{Exp}(s))$. Thus $\mathcal{M}|_w, s$ exists. We now show that $\mathcal{M}|_w, s \leftrightarroweq_o \mathcal{N}|_w, s'$.

Let R be $\{(t,t') \in S_{\mathcal{M}|_w} \times S_{\mathcal{N}|_w} \mid \mathcal{M}, t \leftrightarroweq_o \mathcal{N}, t'\}$. Clearly $(s,s') \in R$. Note that if $\mathcal{L}(\mathit{Exp}(t)) = \mathcal{L}(\mathit{Exp}(t'))$ then $\mathcal{L}(\mathit{Exp}(t) \backslash w) = \mathcal{L}(\mathit{Exp}(t') \backslash w)$; this proves the invariance for observations. Based on this invariance, it is not hard to verify that R is indeed an observation bisimulation between $\mathcal{M}|_w$ and $\mathcal{N}|_w$.

Since $\mathcal{M}|_w, s \leftrightarroweq_o \mathcal{N}|_w, s'$, by induction hypothesis we conclude that

$$\mathcal{M}|_w, s \vDash \neg\psi.$$

Clearly, this contradicts the assumption that $\mathcal{M}, s \vDash [\pi]\psi$.

$[\equiv_{\text{POL}} \implies \leftrightarroweq_o]$: Let $R = \{(t,t') \in S_{\mathcal{M}} \times S_{\mathcal{N}} \mid \mathcal{M}, t \equiv_{\text{POL}} \mathcal{N}, t'\}$. We can show that R is an observation bisimulation. All the conditions are standard and thus can be handled by standard techniques except the new clause about the invariance for observations: we need to show that tRt' implies $\mathcal{L}(\mathit{Exp}(t)) = \mathcal{L}(\mathit{Exp}(t'))$. However, this is trivial, since in the language of POL we can express $\langle w \rangle \top$, so that $M, t \vDash \langle w \rangle \top \iff w \in \mathcal{L}(\mathit{Exp}(t))$. ■

Intuitively, these epistemic expectation models can be seen as compact representations of certain epistemic temporal models [2, 3]. An epistemic temporal model is a Kripke model with both epistemic and temporal binary relations between possible worlds. To make the link more precise, we can relate POL on epistemic expectation models to the same language on epistemic temporal models with the usual PDL-style interpretation of $[\pi]\varphi$ formulas, as we now proceed to show. First let us define the epistemic temporal models that are generated from epistemic expectation models.

Definition 13. *Let $\mathcal{M}$ be an epistemic expectation model $\langle S, \sim_i, V, \mathit{Exp} \rangle$. The $\mathcal{M}$-generated epistemic temporal model (notation:* ET$(\mathcal{M})$*) is defined as $\langle H, \overset{a}{\to}, \sim_i', V' \rangle$ where:*

- $H = \{(s,w) \mid s \in S, w = \epsilon$ *or* $w \in \mathcal{L}(\mathit{Exp}(s))\}$*;*
- $(s,w) \overset{a}{\to} (t,v) \iff s = t$ *and* $v = wa, a \in \mathbf{\Sigma}$*;*
- $(s,w) \sim_i (t,v) \iff s \sim_i t$ *and* $w = v$*;*
- $p \in V'(s,w) \iff p \in V(s)$.

From this definition, it is not hard to see that all the agents can observe all the actions. We can define the semantics of POL formulas on generated epistemic temporal models $\mathcal{N}$ (we only show the non-trivial part):

$$\begin{array}{lcl} \mathcal{N}, h \vDash_{\text{EPDL}} K_i\varphi & \Leftrightarrow & \text{for all } h' : (h \sim_i h' \text{ implies } \mathcal{N}, h' \vDash_{\text{EPDL}} \varphi) \\ \mathcal{N}, h \vDash_{\text{EPDL}} [\pi]\varphi & \Leftrightarrow & \text{for each } w \in \mathcal{L}(\pi), h \xrightarrow{w} h' \text{ implies } \mathcal{N}, h' \vDash_{\text{EPDL}} \varphi \end{array}$$

We call the above semantically defined logic Epistemic-PDL (EPDL): the language of POL interpreted on epistemic temporal models with respect to $\vDash_{\text{EPDL}}$. To establish the precise link between epistemic expectation models and epistemic temporal models, we can prove the following.

Proposition 14. *Given a pointed* POL *model $\mathcal{M}, s$, and a* POL *formula φ, it can be shown that:*

$$\mathcal{M}, s \vDash \varphi \iff \text{ET}(\mathcal{M}), (s, \epsilon) \vDash_{\text{EPDL}} \varphi.$$

Proof. We need to show for any epistemic expectation model $\mathcal{M}, s$ and any POL formula φ:

$$\mathcal{M}, s \vDash \varphi \iff \text{ET}(\mathcal{M}), (s, \epsilon) \vDash_{\text{EPDL}} \varphi$$

We prove this by induction on φ. The Boolean case and the $K_i\psi$ case are trivial. Now consider the case $[\pi]\psi$. Suppose without loss of generality that there is an epistemic expectation model $\mathcal{M}, s \vDash [\pi]\psi$ and $\text{ET}(\mathcal{M}), (s, \epsilon) \nvDash_{\text{EPDL}} [\pi]\psi$. Then there exists a $w \in \mathcal{L}(\pi)$ such that $\text{ET}(\mathcal{M}), (s, w) \nvDash \psi$. By the definition of $\text{ET}(\mathcal{M})$, we conclude that $w \in \textit{Exp}(s)$, thus $\mathcal{M}|_w$ exists. Based on the definition of $\text{ET}(\mathcal{M})$, it is not hard to show that $\text{ET}(\mathcal{M}|_w), (s, \epsilon)$ is bisimilar (with respect to both $\sim$ and $\rightarrow$) to $\text{ET}(\mathcal{M}), (s, w)$. Since EPDL is clearly invariant under bisimulation, we have:

$$\text{ET}(\mathcal{M}|_w), (s, \epsilon) \vDash_{\text{EPDL}} \neg\psi.$$

By induction hypothesis, $\mathcal{M}|_w, s \vDash \neg\psi$, which contradicts the assumption that $\mathcal{M}, s \vDash [\pi]\psi$. ■

Note that the generated epistemic temporal models can be infinite, and thus the above result does not give a straightforward model checking procedure for POL. According to the semantics of $[\pi]\varphi$ we need to check infinitely many $w \in \mathcal{L}(\pi)$. Fortunately, this can be handled by partitioning $\mathcal{L}(\pi)$ into a finite number of regular expressions $\pi_0 \dots \pi_k$ such that for any $0 \leq i \leq k$ and any $w, v \in \mathcal{L}(\pi_i)$, we have $\mathcal{M}|_w = \mathcal{M}|_v$, providing decidability of model checking after all (see [15] for details in a similar setting).

3. Expectation Comes from Protocols

Epistemic expectation models describe the agents' expected observations, which in turn influence their reasoning. We investigate how agents acquire and change their expectations, by looking at protocols and protocol models as sources for the expected observations.

3.1. Protocol expressions

Informally, a protocol is a *rule* telling us what we should do under what conditions. Protocols are ubiquitous in our daily life. A formal way of expressing such protocols or rules is to use a specification language. We specify protocols in the following language of *protocol expressions* $\mathcal{L}_{prot}$:

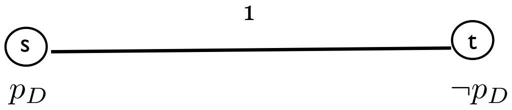
Definition 15 (Protocol expression). *The language $\mathcal{L}_{prot}$ of* protocols *is defined by the following BNF:*

$$\eta \quad ::= \quad \delta \mid \varepsilon \mid a \mid ?\varphi \mid \eta \cdot \eta \mid \eta + \eta \mid \eta^*$$

where δ stands for the empty language $\emptyset$, the constant ε represents the empty string, and $\varphi \in Bool(\mathbf{P})$.

The above language of protocol expressions is obtained by adding Boolean tests to observation expressions. For example, $(?love \cdot stay)^* \cdot (?\neg love \cdot separate)$ expresses 'we should stay together as long as we are in love'. For a discussion on more complicated test scenarios (for example, considering agents' knowledge), see Section 7. We use test conditions in protocol expressions to describe the conditions under which certain observations can happen. A protocol without tests corresponds to observations without any conditions. This is the difference between *protocols* and the *observations that arise out of such protocols*, and we maintain this difference by adding tests to the observation expressions in order to express protocols. In the latter part of this section we will talk about public and private protocols. To this end, we will use dynamic epistemic logic (DEL)-like models to discuss knowledge and ignorance about protocols.

In the story of Example 7, there seems to be an underlying protocol: *if* you are Dutch-related, *then* you kiss three times and *if* you are non-Dutch-related, *then* you kiss two times. This is the reason for the agent to have the corresponding expectations of the observations. This protocol (call it π_K) can be expressed as $?p_D \cdot a \cdot b \cdot a + ?\neg p_D \cdot a \cdot b$. We would like to generate the epistemic expectation model in Example 7 (see page 7) from the protocol π_K and the following epistemic model:



Intuitively, the information of the protocol π_K can be incorporated by adding to each state the possible observations allowed by the protocol. We now move on to the technical details.

To compute the expected observations corresponding to a given protocol, we first define the semantics of protocol expressions. Intuitively, we associate to each protocol η a set $\mathcal{L}_g(\eta)$ of *guarded observations* in the form of

$$\rho_0 a_0 \rho_1 a_1 \dots \rho_k a_k,$$

where each $\rho_i \subseteq \mathbf{P}$ denotes a state of affairs (the atomic propositions $p \in \rho$ are true while the others are false), encoding the conditions for the later observations to happen. For Boolean formulas φ, we write $\rho \vDash \varphi$ if φ is true under ρ (viewed as a valuation: p is true iff $p \in \rho$).

Definition 16. *The set of guarded observations corresponding to a protocol expression is defined by induction, as follows:*

$$\begin{aligned}
&\mathcal{L}_g(\delta) = \emptyset \\
&\mathcal{L}_g(\varepsilon) = \{\rho \mid \rho \subseteq \mathbf{P}\} \\
&\mathcal{L}_g(a) = \{\rho a \rho \mid \rho \subseteq \mathbf{P}\} \\
&\mathcal{L}_g(?\psi) = \{\rho \mid \rho \vDash \psi, \rho \subseteq \mathbf{P}\} \\
&\mathcal{L}_g(\eta_1 \cdot \eta_2) = \{w \diamond v \mid w \in \mathcal{L}_g(\eta_1), v \in \mathcal{L}_g(\eta_2)\} \\
&\mathcal{L}_g(\eta_1 + \eta_2) = \mathcal{L}_g(\eta_1) \cup \mathcal{L}_g(\eta_2) \\
&\mathcal{L}_g(\eta^*) = \{\rho \mid \rho \subseteq \mathbf{P}\} \cup \textstyle\bigcup_{n>0}(\mathcal{L}_g(\eta^n)),
\end{aligned}$$

where $\diamond$ is the fusion product*: $w \diamond v = w'\rho v'$ when $w = w'\rho$ and $v = \rho v'$, and not defined otherwise.*

Note that the ρ_i's in a guarded observation remain unchanged since no *factual change* is introduced by the execution of the actions (see Section 6 for a detailed discussion of fact-changing actions, such as toggling a light switch). We derive the set of observations to be expected under the same condition ρ according to η by a conversion function $f_\rho : \mathcal{L}_{prot} \to \mathcal{L}_{obs}$:

$$
\begin{array}{lllll}
f_\rho(\delta) & = \delta & & & \\
f_\rho(\varepsilon) & = \varepsilon & & f_\rho(\eta \cdot \eta') & = f_\rho(\eta) \cdot f_\rho(\eta') \\
f_\rho(a) & = a & & f_\rho(\eta + \eta') & = f_\rho(\eta) + f_\rho(\eta') \\
f_\rho(?\varphi) & = \begin{cases} \varepsilon & \text{if } \rho \models \varphi \\ \delta & \text{else (i.e., if } \rho \not\models \varphi) \end{cases} & & f_\rho(\eta^*) & = (f_\rho(\eta))^*
\end{array}
$$

Definition 17 (Characteristic formula). *Let $\rho \subseteq \mathbf{P}$. Then we denote by φ_ρ the* characteristic formula *for ρ: $\bigwedge_{p \in \rho} p \wedge \bigwedge_{p \notin \rho} \neg p$. For example, suppose that $\mathbf{P} = \{p, q\}$, then $\varphi_{\{p\}} = p \wedge \neg q$.*

Proposition 18. *(a) For any $\eta \in \mathcal{L}_{prot}$, it holds that*

$\mathcal{L}(f_\rho(\eta)) = \{w \mid w = a_0 \dots a_k,$ *where* $a_i \in \mathbf{\Sigma} \cup \{\varepsilon\}$ *and* $\rho a_0 \rho a_1 \dots a_k \rho \in \mathcal{L}_g(\eta)\}$.

Therefore:
(b) Every η has a normal form η° as follows:

$$\eta^\circ = \sum_{\rho \subseteq \mathbf{P}} (?\varphi_\rho \cdot f_\rho(\eta))$$

such that $\mathcal{L}_g(\eta) = \mathcal{L}_g(\eta^\circ)$. Here φ_ρ is the characteristic formula for ρ as defined in Definition 17.

Proof. We first show (a) by induction on $\eta \in \mathcal{L}_{prot}$. The atomic cases are straightforward. Now we check the complex cases:

$\eta = \eta_1 + \eta_2$:

$$
\begin{array}{rl}
& \mathcal{L}(f_\rho(\eta)) = \mathcal{L}(f_\rho(\eta_1 + \eta_2)) = \mathcal{L}(f_\rho(\eta_1) + f_\rho(\eta_2)) \\
= & \mathcal{L}(f_\rho(\eta_1)) \cup \mathcal{L}(f_\rho(\eta_2)) \\
= & \{w \mid w = a_0 \dots a_k, \text{ and } \rho a_0 \rho \dots \rho a_k \rho \in \mathcal{L}_g(\eta_1)\} \cup \\
& \{w \mid w = a_0 \dots a_k, \text{ and } \rho a_0 \dots a_k \rho \in \mathcal{L}_g(\eta_2)\} \text{(by IH)} \\
= & \{w \mid w = a_0 \dots a_k, \text{ and } \rho a_0 \dots a_k \rho \in \mathcal{L}_g(\eta_1 + \eta_2)\}
\end{array}
$$

$\eta = \eta_1 \cdot \eta_2$:

$$
\begin{array}{rl}
& \mathcal{L}(f_\rho(\eta)) = \mathcal{L}(f_\rho(\eta_1 \cdot \eta_2)) = \mathcal{L}(f_\rho(\eta_1) \cdot f_\rho(\eta_2)) \\
= & \{wv \mid w \in \mathcal{L}(f_\rho(\eta_1)) \text{ and } v \in \mathcal{L}(f_\rho(\eta_2))\} \\
= & \{wv \mid w = c_0 \dots c_m \text{ such that } \rho c_0 \dots c_m \rho \in \mathcal{L}_g(\eta_1) \\
& \text{ and } v = b_0 \dots b_n \text{ such that } \rho b_0 \dots b_n \rho \in \mathcal{L}_g(\eta_2)\} \text{(by IH)} \\
= & \{u \mid u = a_0 \dots a_k, \text{ and } \rho a_0 \dots a_k \rho \in \mathcal{L}_g(\eta_1 \cdot \eta_2)\} \text{(by fusion product)}
\end{array}
$$

$\eta = \eta_1^*$:

$$\begin{array}{rl} & \mathcal{L}(f_\rho(\eta)) = \mathcal{L}(f_\rho(\eta_1^*)) = \mathcal{L}((f_\rho(\eta_1))^*) \\ = & \{\epsilon\} \cup \bigcup_{n>0} \mathcal{L}((f_\rho(\eta_1))^n) \\ = & \{u \mid u = a_0 \dots a_k, \text{ and } \rho a_0 \dots a_k \rho \in \{\rho \mid \rho \subseteq \mathbf{P}\} \cup \bigcup_{n>0} \mathcal{L}_g(\eta_1^n)\} \text{(by IH)} \\ = & \{u \mid u = a_0 \dots a_k, \text{ and } \rho a_0 \dots a_k \rho \in \mathcal{L}_g(\eta_1^*)\} \end{array}$$

This completes the proof for (a). From (a) and the definition of $\mathcal{L}_g$, it follows that:

$$\mathcal{L}_g(f_\rho(\eta)) = \{\rho' a_0 \dots a_k \rho' \mid \rho' \subseteq \mathbf{P} \text{ and } \rho a_0 \dots a_k \rho \in \mathcal{L}_g(\eta)\}.$$

Let $G_\rho^\eta = \{\rho a_0 \rho a_1 \dots a_k \rho \mid \rho a_0 \rho a_1 \dots a_k \rho \in \mathcal{L}_g(\eta)\}$, the set of all ρ-guarded expressions in $\mathcal{L}_g(\eta)$. Then, by fusion product, it follows that $\mathcal{L}_g(?\varphi_\rho \cdot f_\rho(\eta)) = G_\rho^\eta$. Thus,

$$\mathcal{L}_g(\eta^\circ) = \mathcal{L}_g(\sum_{\rho \subseteq \mathbf{P}} (?\varphi_\rho \cdot f_\rho(\eta))) = \bigcup_{\rho \subseteq \mathbf{P}} \mathcal{L}_g(?\varphi_\rho \cdot f_\rho(\eta)) = \bigcup_{\rho \subseteq \mathbf{P}} G_\rho^\eta = \mathcal{L}_g(\eta).$$

This proves (b). ■

From Proposition 18, according to the protocol η, the expected observations on a state s in an epistemic model $\mathcal{M}$ can be computed by $f_{V_\mathcal{M}(s)}(\eta)$. For example, $f_{\{p\}}(?p \cdot a + ?\neg p \cdot b) = a$. However, not every epistemic expectation model can be generated by a single protocol. We will investigate this issue in the next subsection.

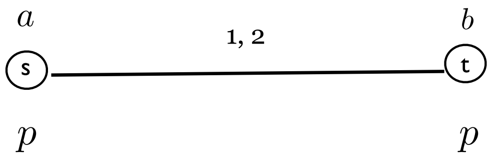
3.2. Protocol models

We introduce *epistemic protocol models* to represent uncertainty about protocols:

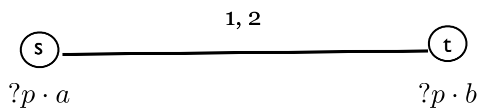
Definition 19 (Epistemic protocol model). *An* epistemic protocol model *$\mathcal{A}$ is a triple $\langle T, \sim, Prot\rangle$, where T is a domain of abstract objects, $\sim$ stands for a set of accessibility (equivalence) relations $\{\sim_i \mid i \in \mathbf{I}\}$, and $Prot : T \to \mathcal{L}_{prot}$ assigns to each domain object a protocol. We call a pointed epistemic protocol model an* epistemic protocol *and a singleton epistemic protocol model (T is singleton) a* public protocol*.*

Note that public protocols are (implicitly) commonly known by all the agents.

Example 20. *Consider the epistemic expectation model:*



We cannot associate a protocol η to the epistemic skeleton of the above model in such a way that $f_{V_M(s)}(\eta) = a$ and $f_{V_{\mathcal{M}}(t)}(\eta) = b$, since $V_{\mathcal{M}}(s) = V_{\mathcal{M}}(t)$. Note that taking $?p(a+b)$ for η does not work. This model represents the uncertainty of the agents about the protocol:



We will now proceed towards our main result in this section, namely that an epistemic observation state uniquely determines an epistemic protocol, and that an epistemic protocol and an epistemic state together uniquely determine an epistemic observation state. To show the correspondence, we need one more semantic operation, that is a *modal product operation* of an epistemic expectation model and a protocol model. It formalizes the change in possible observations induced by a protocol. We should see this definition as installing a new protocol, by means of *novel* observations, into the epistemic expectation model, and thus completely obliterating the *current* expected observations.

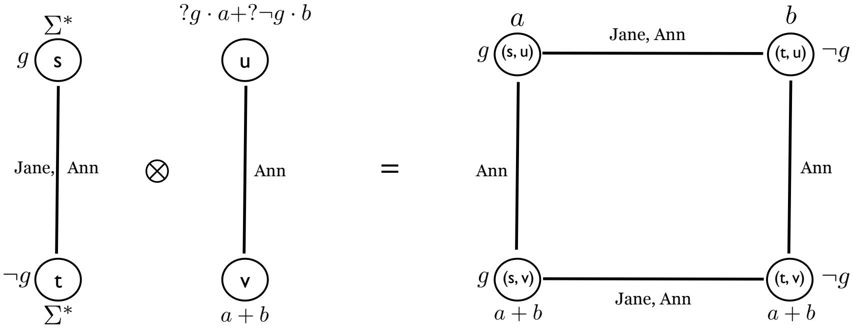
Definition 21 (Protocol update). *Given an epistemic expectation model $\mathcal{M}_{exp} = \langle S, \sim, V, Exp \rangle$ and an epistemic protocol model $\mathcal{A} = \langle T, \sim, Prot \rangle$, we define the product $(\mathcal{M}_{exp} \otimes \mathcal{A}) = (S', \sim', V', Exp')$ as follows:*

- $S' = \{(s,t) \in S \times T : \mathcal{L}(f_{V_{\mathcal{M}}(s)}(Prot(t))) \neq \emptyset\}$*;*
- $(s,t) \sim'_i (s',t')$ *iff* $s \sim_i s'$ *in* $\mathcal{M}_{exp}$ *and* $t \sim_i t'$ *in* $\mathcal{A}$*;*
- $V'(s,t) = V(s)$*;*
- $Exp'((s,t)) = f_{V_{\mathcal{M}}(s)}(Prot(t))$*.*

We mentioned that epistemic models can be seen as special cases of epistemic expectation models, namely with the ‘anything goes’ protocol. Therefore, also in that case the product operation between an epistemic model and a protocol model corresponds to the installation of a protocol.

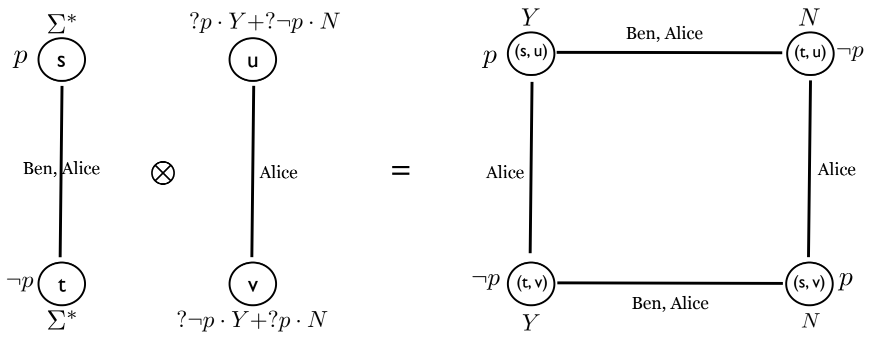
We now illustrate the definition of protocol update by the scenarios presented in Example 1 and Example 2 of the introduction. In the pictures below, we assume reflexivity, symmetry, and transitivity of the accessibility relations.

Example 22. *In the scenario of Example 1, at the beginning neither Jane nor Ann knows the basic proposition g (Kate is gay). However, one of them, Jane, is aware of the protocol that: **if** Kate is gay **then** she will make the statement ‘I am musical, I like Kathleen Ferrier’s voice’ (action a); and **if** she is not gay, **then** she will talk about something else (action b). However, Ann has no idea whether a and b can carry such information. The scenario is modeled as follows, where the last model is the epistemic expectation model resulting from the update of the protocol on the first epistemic model:*



Here, g denotes the fact that ‘Kate is gay’, a denotes the observation of Kate making the ‘musical statement’ and b stands for Kate saying something else.

Example 23. *We now consider the scenario of Example 2. After Carl’s first description of the night of Valentine’s day, Ben and Alice still do not know what has happened. Now, the wink from Carl ‘installs’ the epistemic protocol which creates uncertainty in Alice about the meaning of Ben’s later statements. In contrast, because Ben knows Carl so well, he immediately gets the protocol Carl is using. The modeling is as follows:*



Here, p denotes the fact that 'Something has happened involving Mike and Sara on Valentine's night', while 'Y' corresponds to Carl answering affirmatively to Alice's question, and 'N' to Carl answering negatively.

We assume that Alice's confusion would lead her to consider the possibility of a protocol where Carl would say "Yes" if indeed nothing has actually happened. Because of Carl's wink, however, Alice becomes very distrustful towards him.

According to our definition, an epistemic protocol model acts on an epistemic model, thereby determining a unique epistemic expectation model. In the rest of this section we will investigate the converse question: Can an arbitrary epistemic expectation model be generated by updating an epistemic model by an epistemic protocol model? This is indeed the case, as we will now show.

Proposition 24. *Given an epistemic expectation model $\mathcal{M} = (\mathcal{N}, Exp)$, there is an epistemic model $\mathcal{N}'$ and an epistemic protocol model $\mathcal{A}$ such that $\mathcal{M} \leftrightarroweq_o \mathcal{N}' \otimes \mathcal{A}$.*

Proof. Let $\mathcal{N}' = (S', \sim', V')$ be the universal ignorance model, i.e., $S' = \mathcal{P}(\mathbf{P})$, for each $i, \sim'_i = S' \times S'$, and $V'(\rho) = \rho \subseteq \mathbf{P}$. Given $\mathcal{M} = (S, \sim, V, Exp)$, let $\mathcal{A} = (S, \sim, Prot)$ such that $Prot(s) = ?\varphi_{V(s)} \cdot Exp(s)$. (Remember that $\varphi_{V(s)}$ is the characteristic formula of $V(s) \subseteq \mathbf{P}$, see Definition 17.) Now we show that $\mathcal{M} \leftrightarroweq_o \mathcal{N}' \otimes \mathcal{A}$ by proving that $R = \{(s, (\rho, s)) \mid V(s) = \rho\}$ is a bisimulation relation.

The invariance conditions are immediate. Now suppose $s \sim_i t$ in $\mathcal{M}$, then $(\rho, s) \sim_i (V(t), t)$ in $\mathcal{N}' \otimes A$ by the definition of the product. Obviously, $tR(\rho', t)$, where $\rho' = V(t)$.

Suppose $(\rho, s) \sim_i (\rho', t)$. Then $V(t) = \rho'$. Therefore $s \sim_i t$ and $tR(\rho', t)$.
■

This result shows that every epistemic expectation model is reasonable in the sense that it can be generated from an epistemic model by *some* epistemic protocol model. However, it is more intuitive to consider the particular epistemic model $\mathcal{N}$ in $\mathcal{M} = (\mathcal{N}, \mathit{Exp})$, and ask if there is a protocol model $\mathcal{A}$ such that $\mathcal{N} \otimes \mathcal{A} \leftrightarroweq_o \mathcal{M}$. For singleton protocol models, we have a characterization result. First we need a definition.

Definition 25. *An epistemic expectation model $\mathcal{M}$ is said to be* Boolean normal *if for any two worlds s, t in it, $V_{\mathcal{M}}(s) = V_{\mathcal{M}}(t) \implies \mathcal{L}(\mathit{Exp}(s)) = \mathcal{L}(\mathit{Exp}(t))$.*

Theorem 26. *Given an epistemic expectation model $\mathcal{M} = (\mathcal{N}, \mathit{Exp})$, $\mathcal{M}$ is Boolean normal iff there exists a singleton protocol model $\mathcal{A}$ such that $\mathcal{N} \otimes \mathcal{A} \leftrightarroweq_o \mathcal{M}$, where $\leftrightarroweq_o$ is total.*

Proof. $\Rightarrow$: Let φ_s be the Boolean characterization formula corresponding to $V_{\mathcal{N}}(s)$. Let

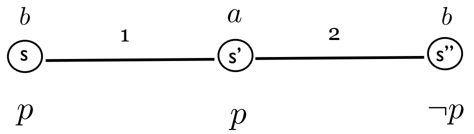
$$\eta_{\mathcal{M}} = \sum_{s \text{ in } \mathcal{N}} ?\varphi_s \cdot \mathit{Exp}(s).$$

Because of the finiteness of $\mathbf{P}$ and Boolean normality, $\eta_{\mathcal{M}}$ has a finite representation. Let $\mathcal{A}_{\eta_{\mathcal{M}}}$ be the singleton pointed protocol model with *Prot* assigning $\eta_{\mathcal{M}}$ to the single point. We can verify that $\mathcal{N} \otimes \mathcal{A}_{\eta_{\mathcal{M}}} \leftrightarroweq_o \mathcal{M}$.

$\Leftarrow$: Suppose $\mathcal{M}$ is not Boolean normal, then there are s, t in $\mathcal{M}$ such that $V(s) = V(t)$ and $\mathit{Exp}(s) \neq \mathit{Exp}(t)$. Due to the normal form of protocols, updating with a public protocol on s, t will result in the same observations. So there cannot be any single pointed protocol model to do the job. ■

Not every epistemic expectation model is Boolean normal, therefore, by Theorem 26, not every epistemic expectation model can be generated by a public protocol on its epistemic skeleton. In fact, as demonstrated by the following example, there are epistemic expectation models which cannot be generated by *any* protocol model on its epistemic skeleton.

Example 27. *Consider the following epistemic expectation model $\mathcal{M}$; we will show that $\mathcal{M}$ cannot be generated by any epistemic protocol on its epistemic skeleton:*



Suppose towards contradiction that there is a protocol model $\mathcal{A}$ such that the execution of $\mathcal{A}$ on the epistemic skeleton of $\mathcal{M}$ gives an epistemic expectation model that is bisimilar to $\mathcal{M}$. To compose s' in the epistemic expectation model, we need a state t in the protocol model such that Prot(t) allows a to happen if p is true. Then t can be composed with the leftmost p-world above as well, since the left world and middle world are Boolean indistinguishable. Therefore there will be a $p(a)$-world in the resulting model which cannot reach any $\neg p$-world in one step, due to the definition of $\otimes$ (the leftmost state above cannot reach any $\neg p$-world in one step).

This leads us to consider a subclass of the epistemic expectation models given as follows.

Definition 28 (Observational saturation). *An epistemic expectation model $\mathcal{M}$ is said to be* observationally saturated *iff the following holds:*
For all states v, s, t in $\mathcal{M}$, for all $i \in \mathbf{I}$: If $v \sim_i s$ and $V(s) = V(t)$, then there exists u in $\mathcal{M}$ such that $v \sim_i u$, $s \leftrightarroweq u$ and Exp$_{\mathcal{M}}(t) =$ Exp$_{\mathcal{M}}(u)$.

Note that every Boolean normal epistemic expectation model $\mathcal{M}$ is observationally saturated: suppose $w \sim_i s$ and $V(s) = V(t)$ then clearly $s \leftrightarroweq s$ and *Exp*$_{\mathcal{M}}(s) =$ *Exp*$_{\mathcal{M}}(t)$ since $\mathcal{M}$ is Boolean normal. Note that the model in Example 27 is not observationally saturated: the leftmost world and the middle world share the same valuation but different observations, however, there is no 1-successor of the leftmost world that is (standard) bisimilar to the leftmost world and has the same expectation as the middle world.

In the following, we show that observational saturation is a sufficient condition for an epistemic expectation model to be generatable from its epistemic skeleton.

Theorem 29. *Given an epistemic expectation model $\mathcal{M} = (\mathcal{N}, Exp)$, if $\mathcal{N}$ is observationally saturated then there is a protocol model $\mathcal{A}$ such that $\mathcal{N} \otimes \mathcal{A} \leftrightarroweq_o \mathcal{M}$.*

Proof. Suppose $\mathcal{N} = (S, \sim, V)$. For any $s \in S$, let $\varphi_s^{\mathcal{N}}$ be the Boolean characteristic formula of s. Let $\mathcal{A} = (S, \sim', \mathit{Prot})$ where $\mathit{Prot}(s) = ?\varphi_s^{\mathcal{N}} \cdot \mathit{Exp}_{\mathcal{M}}(s)$ and $\sim'_i = S \times S$ for each $i \in \mathbf{I}$.

Let $R \subseteq S \times S_{\mathcal{N} \otimes \mathcal{A}}$ be the binary relation

$$\{(w, (v, t)) \mid w \leftrightarroweq v \text{ and } \mathit{Exp}_{\mathcal{M}}(w) = \mathit{Exp}_{\mathcal{M}}(t)\}$$

It is easy to see that $(w, (w, w)) \in R$ for all $w \in S$. We need to show that R is indeed a total observation bisimulation (see Definition 11).

To this end, suppose $wR(v, t)$. Since $\mathit{Prot}(t) = ?\varphi_t^{\mathcal{N}} \cdot \mathit{Exp}_{\mathcal{M}}(t)$ and (v, t) is in $\mathcal{N} \otimes \mathcal{A}$, we have $\mathcal{N}, v \vDash \varphi_t^{\mathcal{N}}$. From the definition of R, we have $\mathit{Exp}_{\mathcal{M}}(w) = \mathit{Exp}_{\mathcal{M}}(t)$ and $w \leftrightarroweq v$ thus w and (v, t) should have the same valuation according to the definition of $\otimes$. Moreover, it holds that $\mathit{Exp}_{\mathcal{M}}(w) = \mathit{Exp}_{\mathcal{N} \otimes \mathcal{A}}((v, t))$ since $\mathit{Exp}_{\mathcal{M}}(w) = \mathit{Exp}_{\mathcal{M}}(t)$ and $\mathit{Prot}(t) = ?\varphi_t^{\mathcal{N}} \cdot \mathit{Exp}_{\mathcal{M}}(t)$. Now we only need to check the Zig-Zag conditions.

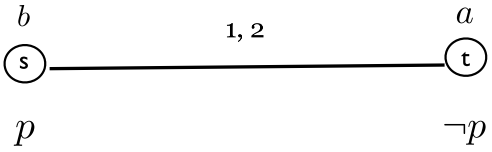
So, suppose $w \sim_i w'$ in $\mathcal{M}$. Since $w \leftrightarroweq v$ there is a v' in $\mathcal{M}$ such that $v \sim_i v'$ and $w' \leftrightarroweq v'$ in $\mathcal{M}$. Therefore $V_{\mathcal{M}}(w') = V_{\mathcal{M}}(v')$, thus (v', w') exists in $\mathcal{N} \otimes \mathcal{A}$. Now due to the fact that the relations in $\mathcal{A}$ are universal, we have $(v, t) \sim_i (v', w')$. It is clear that $(w', (v', w')) \in R$.

Suppose $(v, t) \sim_i (v', t')$ in $\mathcal{N} \otimes A$; then $v \sim_i v'$ in $\mathcal{M}$ and $V_{\mathcal{M}}(v') = V_{\mathcal{M}}(t')$. Since $w \leftrightarroweq v$, there is a w' in $\mathcal{M}$ such that $w \sim_i w'$ and $w' \leftrightarroweq v'$ in $\mathcal{M}$. Therefore $V_{\mathcal{M}}(w') = V_{\mathcal{M}}(v') = V_{\mathcal{M}}(t')$. Now consider w' and t': since $\mathcal{M}$ is observationally saturated, there is a w'' in $\mathcal{M}$ such that $w \sim_i w''$, $w' \leftrightarroweq w''$ and $\mathit{Exp}_{\mathcal{M}}(w'') = \mathit{Exp}_{\mathcal{M}}(t')$. Since $w'' \leftrightarroweq w'$, we have $w'' \leftrightarroweq v'$. Therefore $(w'', (v', t')) \in R$.

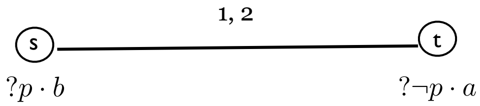
To complete the proof, we need to show that the bisimulation is total. It is clear that for each w: $(w, (w, w)) \in R$. Now for any (v, t) in $\mathcal{N} \otimes \mathcal{A}$, we need to show that (v, t) is linked to *some* world in $\mathcal{M}$ by R. Suppose (v, t) exists in $\mathcal{M}$ then $V_{\mathcal{M}}(v) = V_{\mathcal{M}}(t)$. Note that $v \sim_i v$ for any $i \in \mathbf{I}$ since $\sim_i$ is reflexive. By observational saturation, there is a w in $\mathcal{M}$ such that $v \sim_i w$, $v \leftrightarroweq w$ and $\mathit{Exp}_{\mathcal{M}}(w) = \mathit{Exp}_{\mathcal{M}}(t)$. Therefore $(w, (v, t)) \in R$. ■

3.3. Equivalence of protocols

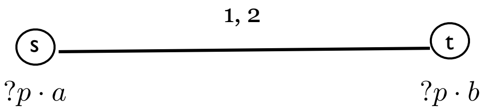
In the introduction, we stated that one epistemic expectation model might be generated in different ways, even based on the same epistemic model. For example, consider the following expectation model:



It can be generated from its epistemic skeleton by updating with the public protocol $?p \cdot b{+}?\neg p \cdot a$ or with the epistemic protocol model:



Actually, on arbitrary epistemic models, the announcement of $?p{\cdot}b{+}?\neg p{\cdot}a$ will always yield the same result as the above epistemic protocol model. On the other hand, the announcement $?p \cdot (a + b)$ gives a different update result on the same epistemic model compared to the update with the following epistemic protocol:



Such examples suggest a notion of equivalence between protocol models.

Definition 30 (Effective equivalence). *Two protocol models $\mathcal{A}$ and $\mathcal{B}$ are said to be* effectively equivalent *(notation: $\mathcal{A} \equiv_{ef} \mathcal{B}$) if for any epistemic expectation model $\mathcal{M}$: $\mathcal{M} \otimes \mathcal{A} \leftrightarroweq_o \mathcal{M} \otimes \mathcal{B}$.*

Inspired by the idea of action emulation introduced by Van Eijck, Ruan and Sadzik in [19] and further explored in [20], we characterize the notion of effective equivalence by the following structural equivalence. To simplify the notation, let $\mathcal{L}^{\rho}(\eta)$ be $\mathcal{L}(f_{\rho}(\eta))$ (cf. Proposition 18).

Definition 31 (Protocol emulation). *Two protocol models $\mathcal{A} = (S, Prot)$ and $\mathcal{B} = (T, Prot)$ are said to be* emulated *(notation: $\mathcal{A} \approx \mathcal{B}$) if there is a binary relation $E \subseteq S \times T$ such that for every $s \in \mathcal{A}$, there exists a $t \in \mathcal{B}$ with sEt, and for every $t \in \mathcal{B}$, there exists an $s \in \mathcal{A}$ with sEt, and whenever sEt we have that:*

- *there exists $\rho \subseteq \mathbf{P}$ such that $\mathcal{L}^{\rho}(\mathit{Prot}(t)) = \mathcal{L}^{\rho}(\mathit{Prot}(s))$.*
- *if $s \sim_i s'$ in $\mathcal{A}$ then there is a set $T' \subseteq T$ such that:*
  1. *for any $t' \in T'$: $t \sim_i t'$;*
  2. *for any $t' \in T'$: $s'Et'$;*
  3. *for any $\rho \subseteq \mathbf{P}$ such that $\mathcal{L}^{\rho}(\mathit{Prot}(s')) \neq \emptyset$ there exists $t' \in T'$ such that $\mathcal{L}^{\rho}(\mathit{Prot}(s')) = \mathcal{L}^{\rho}(\mathit{Prot}(t'))$*
- *if $t \sim_i t'$ in $\mathcal{B}$ then there is a set $S' \subseteq S$ such that:*
  1. *for any $s' \in S'$: $s \sim_i s'$;*
  2. *for any $s' \in S'$: $s'Et'$;*
  3. *for any $\rho \subseteq \mathbf{P}$ such that $\mathcal{L}^{\rho}(\mathit{Prot}(t')) \neq \emptyset$ there exists $s' \in S'$ such that $\mathcal{L}^{\rho}(\mathit{Prot}(s')) = \mathcal{L}^{\rho}(\mathit{Prot}(t'))$*

When restricted to public protocols, it is not hard to see that $\eta \approx \eta' \iff \mathcal{L}_g(\eta) = \mathcal{L}_g(\eta')$. In general, we have the following result.

Theorem 32. *For all finite protocol models $\mathcal{A}$ and $\mathcal{B}$: $\mathcal{A} \equiv_{ef} \mathcal{B} \iff \mathcal{A} \approx \mathcal{B}$.*

Proof. $\Leftarrow$: Suppose $\mathcal{A} \approx \mathcal{B}$. We need to show for any epistemic expectation model $\mathcal{M}$ that: $\mathcal{M} \otimes \mathcal{A} \leftrightarroweq_o \mathcal{M} \otimes \mathcal{B}$. We define a binary relation between $\mathcal{M} \otimes \mathcal{A}$ and $\mathcal{M} \otimes \mathcal{B}$ as $(w, s)R(v, t) \iff w = v, sEt$ and $\mathit{Exp}((w, s)) = \mathit{Exp}((v, t))$. Whenever $(w, s) \in \mathcal{M} \otimes \mathcal{A}$, $(w, t) \in \mathcal{M} \otimes \mathcal{B}$ for some $t \in \mathcal{B}$. This happens due to the fact that $\mathcal{A} \approx \mathcal{B}$, and the epistemic relations in each model are reflexive. Thus we have that the definition of R is both sound and total. Now we verify the condition Zig of Definition 11 (the invariance condition is trivial by definition of R). Suppose $(w, s) \sim_i (w', s')$, then $w \sim_i w'$ in $\mathcal{M}$ and $s \sim_i s'$ in $\mathcal{A}$. Since sEt, there is a t' in $\mathcal{B}$ such that $t \sim_i t'$, $s'Et'$, and $\mathcal{L}^{\rho_0}(\mathit{Prot}(s')) = \mathcal{L}^{\rho_0}(\mathit{Prot}(t'))$, where $\rho_0 = V(w')$. Clearly (w', t') is in $\mathcal{M} \otimes \mathcal{B}$ and $\mathit{Exp}((w', t')) = \mathit{Exp}((w, s'))$. Thus we have that $(w, t) \sim_i (w', t')$ and $(w', s')R(w', t')$. The condition Zag can be proved in a similar way.

$\Rightarrow$: Suppose $\mathcal{A} \equiv_{ef} \mathcal{B}$. It is clear that for a universal ignorance model $\mathcal{M}$ (cf. the proof of Proposition 24), we have $\mathcal{M} \otimes \mathcal{A} \leftrightarroweq_o \mathcal{M} \otimes \mathcal{B}$. We define a relation E between the state spaces of $\mathcal{A}$ and $\mathcal{B}$ as: sEt iff $(w, s) \leftrightarroweq_o (w, t)$ for some w. We can verify that E is a protocol emulation relation. The first (consistency) condition of protocol emulation is immediate according to the invariance condition of observation bisimulation. Now we show the second one. Suppose $s \sim_i s'$ and sEt. Now consider an arbitrary $\rho \subseteq \mathbf{P}$ such that $\mathcal{L}^\rho(Prot(s')) \neq \emptyset$. Since $\mathcal{M}$ is a universal ignorance model, there is a state w' in $\mathcal{M}$ such that $V(w') = \rho$ and $(w, s) \sim_i (w', s')$. Since sEt then by definition of E, $(w, s) \leftrightarroweq_o (w, t)$. Thus there is a (v', t') in $\mathcal{M} \otimes \mathcal{B}$ such that $(w, t) \sim_i (v', t')$ and $(w', s') \leftrightarroweq_o (v', t')$; clearly w' and v' share the same valuation, thus $w' = v'$ since $\mathcal{M}$ is a universal ignorance model. It follows that $t \sim_i t'$ and $\mathcal{L}^\rho(\mathit{Prot}(s')) = \mathcal{L}^\rho(\mathit{Prot}(t'))$. Thus for all $\rho \subseteq \mathbf{P}$ such that $\mathcal{L}^\rho(Prot(s')) \neq \emptyset$ there is a state t' with $t \sim_i t'$ in $\mathcal{B}$, such that $s'Et'$ and $\mathcal{L}^\rho(\mathit{Prot}(s')) = \mathcal{L}^\rho(\mathit{Prot}(t'))$. The third condition can be shown similarly. The emulation relation is total, as we are considering total bisimulation here. ■

We now extend the framework of POL to provide a DEL-style logical language that can describe the ‘installation’ or ‘change’ of protocols, together with the effect of the observations of agents, based on the current protocol. Note that installing a protocol is different from executing a protocol: Installing a protocol gives the knowledge of the protocol before its execution.

3.4. Epistemic Protocol Logic

In the language of epistemic protocol logic (EPL), we consider protocol models as primitives in the language, giving a DEL-like language.

Definition 33 (Language of EPL). *The* formulas φ *of* EPL *are given by:*

$$\varphi \quad ::= \quad \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid K_i\varphi \mid [\pi]\varphi \mid [!\mathcal{A}_e]\varphi$$

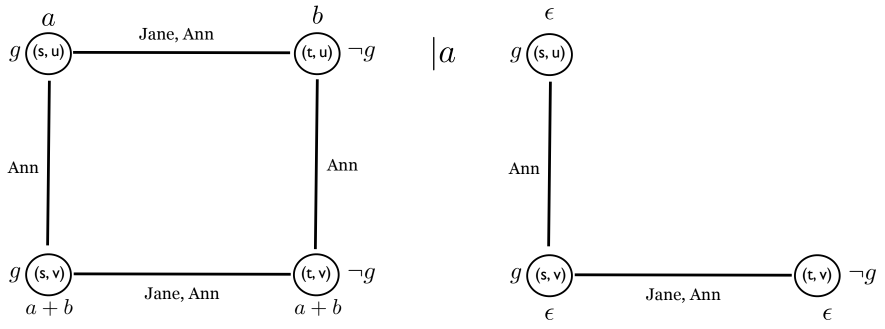
where $p \in \mathbf{P}$, $i \in \mathbf{I}$, $\pi \in \mathcal{L}_{obs}$, and $\mathcal{A}_e$ is an epistemic protocol with the designated state e.

In defining the language we restrict ourselves to finite protocol models. The models for the logic EPL are taken to be the epistemic expectation models $\mathcal{M} = \langle S, \sim, V, \mathit{Exp} \rangle$. The truth definition is given as follows:

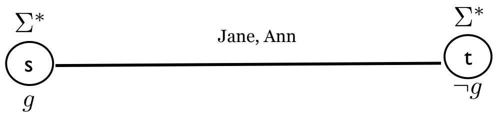
Definition 34 (Truth definition for EPL). *Given an epistemic expectation model $\mathcal{M} = \langle S, \sim, V, \mathit{Exp} \rangle$, a state $s \in S$, and an* EPL*-formula φ, the truth conditions of φ at s coincide with* POL *for the formulas that they have in common. The truth condition for the new formula in* EPL *is defined as follows:*

$$\boxed{\mathcal{M}, s \vDash [!\mathcal{A}_e]\varphi \quad \Leftrightarrow \quad \textit{If } \mathcal{L}(f_{V(s)}(\textit{Prot}(e))) \neq \emptyset \textit{ then } \mathcal{M} \otimes \mathcal{A}, (s, e) \vDash \varphi}$$

Recalling the meaning of the modal product operation, the expression '$[!\mathcal{A}_e]\varphi$' therefore stands for 'after installing the new epistemic protocol $\mathcal{A}_e$, the formula φ is true'. As an example, let us give the model of Example 1 from the introduction, the epistemic expectation model induced by the epistemic protocol (modelled on page 18, call it $\mathcal{A}_e$), and the updated model according to observation a (in the picture, visualized by $|_a$):



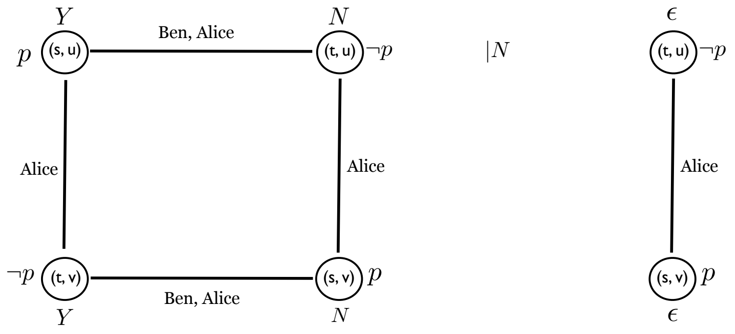
Recall the original model $\mathcal{M}$:



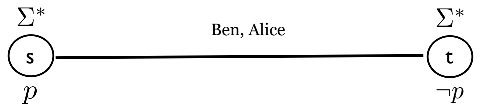
Now we can verify for the actual state s:

$\mathcal{M}, s \vDash [!\mathcal{A}_e][a](K_{Jane}g \wedge \neg K_{Ann}g)$, and
$\mathcal{M}, s \vDash [!\mathcal{A}_e][a]\neg K_{Ann}(K_{Jane}g \vee K_{Jane}\neg g)$.

The picture corresponding to Example 2 from the introduction is as follows. Here, $\mathcal{A}'_{e'}$ is the corresponding epistemic protocol modelled on page 18:



Recall the initial model $\mathcal{N}$:



Now we can verify for the actual state t:

$\mathcal{N}, t \vDash [!\mathcal{A}'_{e'}][N](K_{Ben}\neg p \wedge \neg K_{Alice}\neg p)$, but
$\mathcal{N}, t \vDash [!\mathcal{A}'_{e'}][N]K_{Alice}(K_{Ben}p \vee K_{Ben}\neg p)$.

4. Incorporating factual changes

So far, we have only presented information changing actions, not fact-changing actions: recall that $\mathcal{L}_g(\eta)$ consists of guarded strings with uniform guards only. This may not be so realistic in practice, since many actions used in protocols also change the facts, for example, 'turn on the light if you see that the light is off'. Factual change can be modelled by assigning to each action a function that changes the valuation of basic propositions (as in [21, 22]). Let us now show how protocols based on fact-changing actions can be incorporated in our setting. Following [21], we first introduce *fact-changing actions*.

Definition 35 (Fact-changing actions). *A set of fact-changing actions (fc-actions) is a tuple* $(\mathbf{\Sigma}, \iota)$ *such that* $\iota : \mathbf{\Sigma} \times \mathbf{P} \to Bool(\mathbf{P})$.

Intuitively, ι captures the post-condition of actions: after executing action $a \in \mathbf{\Sigma}$, the propositional atom p is assigned the truth value of the proposition $\iota(a, p)$. Thus, the new truth value of p is the truth value of $\iota(a, p)$

evaluated before executing a. Note that in this paper we restrict $\iota(a, p)$ to be Boolean. For example, let p be the proposition denoting 'the door is closed' and let a be the action 'slam the door'. Then slamming the door (a) has a post-condition given by $\iota(a, p) = \top$. On the other hand, toggling the switch (b) has the post-condition modelled by $\iota(b, q) = \neg q$ if q expresses that 'the switch is on'. Clearly, non-fact-changing actions can be seen as $(\mathbf{\Sigma}, \iota_0)$, where for any $a \in \mathbf{\Sigma}$, $\iota_0(a)$ is the identity function.

For the ease of reading in proofs, we introduce *factual change systems* as an alternative way of representing fact-changing actions. In the following, $\rho \vDash \varphi$ means that the valuation represented by ρ, a subset of $\mathbf{P}$, makes the Boolean formula φ true.

Definition 36 (Factual change system). *A $\mathbf{\Sigma}$-factual change system (fc-system) $\mathcal{F}$ is a tuple (Q, r) where $Q = \mathcal{P}(\mathbf{P})$ and $r : Q \times \mathbf{\Sigma} \to Q$ is a function.*

Because r is a deterministic transition function, it can be extended to the domain of $Q \times \mathbf{\Sigma}^*$ in such a way that $r(\rho, a_0 \cdots a_k)$ is the unique state $\rho' \subseteq \mathbf{P}$ of the fc-system that is reachable from $\rho \subseteq \mathbf{P}$ via transitions sequentially labelled by actions $a_0, \dots, a_k$.

Intuitively, a factual change system explicitly represents the post-conditions of actions that can change the facts on states. We say that a set of fact-changing actions $(\mathbf{\Sigma}, \iota)$ is equivalent to a $\mathbf{\Sigma}$-factual change system (Q, r) if for any $a \in \mathbf{\Sigma}$ and any $\rho, \rho' \subseteq \mathbf{P}$ the following holds:

$$\rho \vDash \bigwedge_{p \in \rho'} \iota(a, p) \wedge \bigwedge_{p \notin \rho'} \neg \iota(a, p) \iff r(\rho, a) = \rho'$$

As a reminder, for each $\rho \subseteq \mathbf{P}$, we write φ_ρ as the abbreviation of the characteristic formula (see Definition 17). Now we show that sets of fact-changing actions can be seen as factual change systems and vice versa.

Proposition 37. *(a) For each set of fc-actions $(\mathbf{\Sigma}, \iota)$ there is an equivalent $\mathbf{\Sigma}$-fc-system.*
(b) For each $\mathbf{\Sigma}$-fc-system there is an equivalent set of fc-actions $(\mathbf{\Sigma}, \iota)$.

Proof. (a) To define the corresponding transition function r in the factual change system, we do the following. For every $\rho \subseteq \mathbf{P}$ and $a \in \mathbf{\Sigma}$, we define $r(\rho, a) = \rho'$ if and only if $\rho \vDash \bigwedge_{p \in \rho'} \iota(a, p) \wedge \bigwedge_{p \notin \rho'} \neg \iota(a, p)$.
(b) For the second part, we can define a set of fact-changing actions $(\mathbf{\Sigma}, \iota)$

by letting $\iota(a,p) = \bigvee_{\rho\subseteq\mathbf{P}}\{\varphi_\rho \mid p \in r(\rho,a)\}$. We need to verify the equivalence condition. Suppose $r(\rho_1,a) = \rho_2$, then by the definition of ι, it is clear that $\rho_1 \vDash \bigwedge_{p\in\rho_2}\iota(a,p)$. Since fc-systems are deterministic, for each $p \notin \rho_2$: $\rho_1 \notin \{\rho \mid p \in r(\rho,a)\}$. Therefore for each $p \notin \rho_2$: $\rho_1 \vDash \neg\bigvee_{\rho\subseteq\mathbf{P}}\{\varphi_\rho \mid p \in r(\rho,a)\}$. Thus $\rho_1 \vDash \bigwedge_{p\notin\rho_2}\neg\iota(a,p)$.

On the other hand, if $r(\rho_1,a) = \rho_3 \neq \rho_2$ then there is a proposition $p \in \mathbf{P}$ on which ρ_2 and ρ_3 do not agree. Suppose that $p \in \rho_3$ but $p \notin \rho_2$. Since $\iota(a,p) = \bigvee_{\rho\subseteq\mathbf{P}}\{\varphi_\rho \mid p \in r(\rho,a)\}$ then $\rho_1 \nvDash \iota(a,p)$. Thus $\rho_1 \nvDash \bigwedge_{p\in\rho_2}\iota(a,p)$. Similarly, we can show that if $p \in \rho_2$ but $p \notin \rho_3$ then $\rho_1 \nvDash \bigwedge_{p\notin\rho_2}\neg\iota(a,p)$. Therefore $\rho_1 \nvDash \bigwedge_{p\in\rho_2}\iota(a,p) \wedge \bigwedge_{p\notin\rho_2}\neg\iota(a,p)$. ■

In the sequel, we only work with fc-systems in the proofs. To interpret observation expressions with respect to an fc-system $\mathcal{F}$, we only need to revise Definition 16 of $\mathcal{L}_g$ as follows:

$$\mathcal{L}_g^{\mathcal{F}}(a) = \{\rho a\rho' \mid \rho \xrightarrow{a} \rho' \text{ in } \mathcal{F}\}$$

To install protocols with factual change on an epistemic model, we need to compute the state-dependent expectations according to those protocols. However, it is not immediately clear how we can rewrite a protocol into a normal form as in Proposition 18, where the tests only happen at the beginning. To model the updates of protocols with factual change, we first need to prove an analogue of Proposition 18. This will be Proposition 41. To prove this proposition we need techniques for guarded automata developed in [23].

Given $\mathbf{P}$, let $\mathbf{T}$ be the set $2^{2^{\mathbf{P}}}$. Intuitively, $X \in \mathbf{T}$ represents a Boolean formula over $\mathbf{P}$.

Definition 38 (Automata on guarded strings [23]). *A finite* automaton on guarded strings *(or a* guarded automaton*) over a finite set of actions $\mathbf{\Sigma}$ and a finite set of atomic propositions $\mathbf{P}$ is a tuple $\mathtt{A} = (Q, \mathbf{\Sigma}, \mathbf{P}, q_0, \mapsto, F)$, where Q is a set of states with the designated start state q_0; $\mapsto$ is a set of transitions labelled by actions in $\mathbf{\Sigma}$ (action transitions) and sets $X \in \mathbf{T}$ (test transitions); F is the set of final states.* $\mathtt{A}$ accepts *a finite string w over $\mathbf{\Sigma} \cup \mathbf{T}$ (notation: $w \in \mathcal{L}_{\mathbf{\Sigma}\cup\mathbf{T}}(\mathtt{A})$), if it accepts w as a standard finite automaton over label set* $\mathbf{\Sigma} \cup \mathbf{T}$. *The acceptance for guarded strings is defined based on the acceptance of normal strings and the following transformation function G which takes a string over $\mathbf{\Sigma} \cup \mathbf{T}$ and outputs a set of guarded strings, as follows:*

$$\begin{array}{rcl} G(a) & = & \{\rho a \rho' \mid \rho, \rho' \subseteq \mathbf{P}\} \\ G(X) & = & \{\rho \mid \rho \in X\} \\ G(ww') & = & \{v\rho v' \mid v\rho \in G(w) \textit{ and } \rho v' \in G(w')\} \end{array}$$

We say that A accepts *a finite guarded string $v : \rho_0 a_0 \rho_1 \ldots a_{k-1}\rho_k$ over $\mathbf{\Sigma}$ and $\mathbf{P}$, if $v \in G(w)$ for some string $w \in \mathcal{L}_{\mathbf{\Sigma}\cup\mathbf{T}}(\mathtt{A})$. Let $\mathcal{L}_g(\mathtt{A})$ be the language of guarded strings accepted by* A.

A guarded automaton is said to be *deterministic* if it satisfies the following properties (cf. [23]):

- Each state is either a state that only has outgoing action transitions (*action state*) or a state that only has outgoing test transitions (*test state*).
- The outgoing action transitions are deterministic: for each action state q and each $a \in \mathbf{\Sigma}$, state q has one and only one a-successor.
- The outgoing test transitions are deterministic: they are labelled by $\{\{\rho\} \mid \rho \subseteq \mathbf{P}\}$ and for each test state q and each ρ, state q has one and only one $\{\rho\}$-successor. Clearly these tests ρ at a test state are logically pairwise exclusive and altogether exhaustive (viewing ρ as the characteristic Boolean formula φ_ρ, see Definition 17).
- The start state q_0 is a test state and all accept states are action states.
- Each cycle contains at least one action transition.

A Kleene-like theorem about the relation between guarded automata and guarded regular expressions has been proved in [23]. Here follows a reminder.

Theorem 39 ([23]). *For each guarded regular expression η over $\mathbf{P}$ and $\mathbf{\Sigma}$ there is a deterministic guarded automaton* A *over $\mathbf{P}$ and $\mathbf{\Sigma}$ such that $\mathcal{L}_g(\eta) = \mathcal{L}_g(\mathtt{A})$, and vice versa.*

Given an fc-system $\mathcal{F}$, we define a translation $t^{\mathcal{F}} : \mathcal{L}_{prot} \to \mathcal{L}_{prot}$ by replacing each a with $\sum_{\rho\subseteq\mathbf{P}}\{?\varphi_\rho \cdot a \cdot ?\varphi_{\rho'} \mid \rho \xrightarrow{a} \rho' \text{ in } \mathcal{F}\}$. It is not hard to see that for each guarded expression η: $\mathcal{L}_g^{\mathcal{F}}(\eta) = \mathcal{L}_g(t^{\mathcal{F}}(\eta))$. From Theorem 39, we have the following corollary.

Corollary 40. *Given an fc-system $\mathcal{F}$, for each guarded expression η, there is a deterministic guarded automaton* $\mathtt{A}$ *and a deterministic finite automaton* $\mathtt{A}'$ *over the alphabet $\mathbf{\Sigma} \cup 2^{\mathbf{P}}$ such that:*

$$\mathcal{L}_g^{\mathcal{F}}(\eta) = \mathcal{L}_g(\mathtt{A}) = \mathcal{L}(\mathtt{A}').$$

Proof. Consider $t^{\mathcal{F}}(\eta)$. The existence of the deterministic guarded automaton $\mathtt{A}$ follows from Theorem 39 directly. By the definition of determinism, $\mathcal{L}(\mathtt{A})$ is a set of guarded strings in the shape of $\{\rho_0\}a_0\{\rho_1\}\cdots\{\rho_{n-1}\}a_{n-1}\{\rho_n\}$. Clearly

$$G(\{\rho_0\}a_0\{\rho_1\}\cdots\{\rho_{n-1}\}a_{n-1}\{\rho_n\}) = \rho_0 a_0 \rho_1 \cdots \rho_{n-1} a_{n-1} \rho_n$$

Now we can build the desired deterministic finite automaton $\mathtt{A}'$ over the symbol set $\mathbf{\Sigma} \cup 2^{\mathbf{P}}$ by simply replacing the transition labels $\{\rho\}$ in $\mathtt{A}$ by ρ. ■

Finally, we are ready to prove an analogue of Proposition 18: there is a normal form of guarded regular expressions with respect to an fc-system $\mathcal{F}$ in which tests only appear at the beginning. This is stated formally in the following proposition.

Proposition 41 (Normal form with respect to $\mathcal{F}$). *Given an fc-system $\mathcal{F}$, every η has a normal form*

$$\eta^{\mathcal{F}} = \sum_{\rho \subseteq \mathbf{P}} (?\varphi_\rho \cdot \pi_\rho)$$

for some $\pi_\rho \in \mathcal{L}_{obs}$ such that $\mathcal{L}_g^{\mathcal{F}}(\eta) = \mathcal{L}_g^{\mathcal{F}}(\eta^{\mathcal{F}})$.

Proof. From Corollary 40, for a given fc-system $\mathcal{F}$ and a guarded expression η we have a deterministic automaton $\mathtt{A}$ over $\mathbf{\Sigma} \cup 2^{\mathbf{P}}$ such that $\mathcal{L}(\mathtt{A}) = \mathcal{L}_g^{\mathcal{F}}(\eta)$. Due to the construction of $\mathtt{A}$, the start state has only outgoing ρ transitions for each $\rho \subseteq \mathbf{P}$, thus we can separate the automaton that corresponds to the guarded regular expression into $|2^{\mathbf{P}}|$ zones. Let q_ρ be the state that is the ρ-successor of the start state in $\mathtt{A}$; by determinism there is only one such state. Let $\mathtt{A}_\rho$ be the ϵ-non-deterministic automaton over $\mathbf{\Sigma}$ just like $\mathtt{A}$, but setting q_ρ as the start state and replacing any label $\rho \subseteq \mathbf{P}$ by ϵ. By Kleene's theorem, there is a regular expression π_ρ over $\mathbf{\Sigma}$ such that $\mathcal{L}(\pi_\rho) = \mathcal{L}(\mathtt{A}_\rho)$. We claim the following:

Claim

$$\mathcal{L}_g^{\mathcal{F}}(\rho \cdot \pi_\rho) = \{\rho v \mid \rho v \in \mathcal{L}_g^{\mathcal{F}}(\eta)\}.$$

Proof. First suppose that $\rho a_0 \rho_1 \cdots \rho_{n-1} a_{n-1} \rho_n \in \mathcal{L}_g^{\mathcal{F}}(\rho \cdot \pi_\rho)$, then $a_0 \ldots a_{n-1} \in \mathcal{L}(\mathtt{A}_\rho)$. Therefore $\rho a_0 \rho'_1 \cdots \rho'_{n-1} a_{n-1} \rho'_n \in \mathcal{L}_g^{\mathcal{F}}(\eta)$ for some $\rho'_1 \ldots \rho'_n$. Since the fc-system is deterministic, $\rho'_i = \rho_i$ for $1 \leq i \leq n$. Thus $\rho a_0 \rho_1 \cdots \rho_{n-1} a_{n-1} \rho_n \in \mathcal{L}_g^{\mathcal{F}}(\eta)$. For the other direction, suppose that $\rho a_0 \rho_1 \cdots \rho_{n-1} a_{n-1} \rho_n \in \mathcal{L}_g^{\mathcal{F}}(\eta)$, then $a_0 \ldots a_{n-1} \in \mathcal{L}(\mathtt{A}_\rho) = \mathcal{L}(\pi_\rho)$. By determinism of $\mathcal{F}$ it is clear that $\rho a_0 \rho_1 \cdots \rho_{n-1} a_{n-1} \rho_n \in \mathcal{L}_g^{\mathcal{F}}(\rho \cdot \pi_\rho)$.

From the claim, we can generate the desired normal form for η with respect to a given fc-system $\mathcal{F}$. ■

Based on Proposition 41, we can define the 'installation' of protocols with fact-changing actions on epistemic expectation models, similar to Definition 21. Before we proceed to the definition, note that given $\eta^{\mathcal{F}} = \sum_{\rho \subseteq \mathbf{P}} (?\varphi_\rho \cdot \pi_\rho)$, we have that $f_\rho(\eta^{\mathcal{F}}) = \pi_\rho$ for any $\rho \subseteq \mathbf{P}$ (cf. the definition of f_ρ before Definition 17). Let us now see how the fact-changing actions affect our knowledge state in this evolving world. To this end we first introduce fact-changing epistemic expectation models and protocol models, $\mathcal{M}_{exp}^{\mathcal{F}}$ and $\mathcal{A}^{\mathcal{F}}$, given by $\langle \mathcal{M}_{exp}, \mathcal{F} \rangle$ and $\langle \mathcal{A}, \mathcal{F} \rangle$, where $\mathcal{M}_{exp}$ is an epistemic expectation model, $\mathcal{A}$ is a protocol model and $\mathcal{F}$ is a factual change system.

Definition 42 (Protocol update with factual changes). *Given a fact-changing epistemic expectation model $\mathcal{M}_{exp}^{\mathcal{F}} = \langle S, \sim, V, Exp, \mathcal{F} \rangle$, and a fact-changing epistemic protocol model $\mathcal{A}^{\mathcal{G}} = \langle T, \sim, Prot, \mathcal{G} \rangle$, we define the product $(\mathcal{M}_{exp}^{\mathcal{F}} \otimes \mathcal{A}^{\mathcal{G}}) = (S', \sim', V', Exp', \mathcal{F}')$ as follows:*

- $S' = \{(s,t) \in S \times T : \mathcal{L}(f_{V_{\mathcal{M}}(s)}(Prot^{\mathcal{G}}(t))) \neq \emptyset\}$;
- $(s,t) \sim'_i (s',t')$ *iff* $s \sim_i s'$ *in* $\mathcal{M}_{exp}$ *and* $t \sim_i t'$ *in* $\mathcal{A}$;
- $V'(s,t) = V(s)$;
- $Exp'((s,t)) = f_{V_{\mathcal{M}}(s)}(Prot^{\mathcal{G}}(t))$;
- $\mathcal{F}' = \mathcal{G}$.

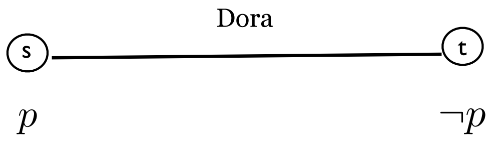
where $Prot^{\mathcal{G}}(t)$ is the normal form of $Prot(t)$ with respect to $\mathcal{G}$.

Accordingly, the truth condition of the new formulas of EPL with respect to these models is changed to the following:

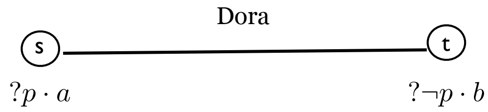
$$\boxed{\begin{array}{rcl} \mathcal{M}^{\mathcal{F}}_{exp}, s \vDash [!\mathcal{A}^{\mathcal{G}}_{e}]\varphi & \Leftrightarrow & \text{If } \mathcal{L}(f_{V(s)}(\mathit{Prot}^{\mathcal{G}}(e))) \neq \emptyset \text{ then } \mathcal{M}^{\mathcal{F}}_{exp} \otimes \mathcal{A}^{\mathcal{G}}, (s,e) \vDash \varphi \\ \mathcal{M}^{\mathcal{F}}_{exp}, s \vDash [\pi]\varphi & \Leftrightarrow & \text{for each } w \in \mathcal{L}(\pi) : (w \in \mathit{init}(\mathit{Exp}(s)) \text{ implies } \mathcal{M}^{\mathcal{F}}_{exp}|_{w}, s \vDash \varphi) \end{array}}$$

where $\mathcal{M}^{\mathcal{F}}_{exp}|_w = (S', \sim', V', \mathit{Exp}', \mathcal{F})$ with $S', \sim', \mathit{Exp}'$ defined as before in $\mathcal{M}|_w$ (cf. Definition 8) and $V'(s) = r(V(s), w)$ where r is the (extended) transition function in $\mathcal{F}$ (cf. Definition 36).

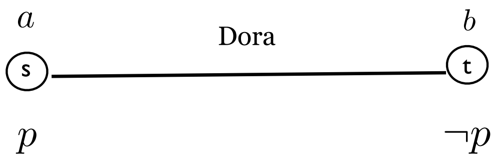
Example 43. *Consider a room where a child is playing with a small plastic seat, and Dora standing outside the room. Before Dora enters, she does not have any idea whether the seat is in an upright position. This is modelled by considering the epistemic model $\mathcal{M}$:*



Here, p stands for 'the seat is in an upright position'. Suppose a denotes the action 'pulling the seat down' and b denotes the action 'pulling the seat up'. Then what the child is doing can be described by the protocol model $\mathcal{A}^{\mathcal{F}}$:



Here, both a and b are fact-changing actions: $\iota(a,p) = \neg p$, and $\iota(b,p) = \neg p$. We note that any epistemic model is an epistemic expectation model which in turn can be considered as a fact-changing epistemic expectation model where ι is the identity mapping in the second argument. The updated product model will be of the form:



At the actual state s of the epistemic model $\mathcal{M}$, we have:

$$\mathcal{M}, s \models [!\mathcal{A}_e^{\mathcal{F}}][a]K_{Dora}\neg p$$

That is, after entering the room, upon observing action a, Dora will come to know that the seat is not in an upright position.

In the following section, we describe a more detailed application of factual change systems.

5. Application: One Hundred Prisoners and a Lightbulb

In this section we model within our framework the '100 prisoners and a lightbulb' puzzle [24, 25] from the novel perspective of the guard in the puzzle. The following description is based on [24].

> A group of 100 prisoners, all together in the prison dining area, are told that they will be all put in isolation cells and then will be interrogated one by one in a room containing a light with an on/off switch. The prisoners may communicate with one another by toggling the light-switch (and that is the only way in which they can communicate). All the prisoners know that the light is initially switched off. There is no fixed order of interrogation, or fixed interval between interrogations, and at any stage every prisoner will be interrogated again sometime in the future. When interrogated, a prisoner can either do nothing, or toggle the light-switch, or announce that all prisoners have been interrogated. If that announcement is true, the prisoners will (all) be set free, but if it is false, they will all be executed. While still in the dining room, and before the prisoners go to their isolation cells, can the prisoners agree on a protocol that will set them free?

Two protocols to solve the puzzle are as follows [24]. We move to the perspective of $n+1$ prisoners, where $n \geq 2$. (The case $n = 1$ is a tricky boundary case which requires special treatment. For simplicity we leave it out in this paper.)

> **Protocol 1** The $n+1$ prisoners appoint one amongst them as the *leader*. The remaining n prisoners are the *followers*. All n followers turn the light on (i.e., toggle the switch) the first time they enter the room when the light is off; on other occasions, they do not toggle the switch. The leader turns off the light (toggles the switch) the first n times that the light is on when he enters the interrogation room; on other occasions, he does not toggle the switch. After turning the light off for the nth time, the leader announces that all prisoners have been interrogated.
>
> **Protocol 2** The leader does exactly as in Protocol 1. The followers do all they do in Protocol 1, but also do more. Each follower counts the number of times the state of the light has changed from *off* to *on* according to his own observation (see the explanation below). If a follower has observed n such changes, he announces that all prisoners have been interrogated.

We say that a follower observes a change of the state of light from *off* to *on*, if the light was off in his last interrogation but the light is on in his current interrogation. Moreover, there are also two special cases in the counting of such changes:

1. Since initially the light is switched off, when a follower enters the room for the first time and observes that the light is on, it counts as an off-on change;
2. When a follower is about to toggle the light from off to on according to the protocol, it also counts as an off-on change.

The above explanation will be made more precise in the formalization of Protocol 2 below.

Note that the interest of Protocol 2 is that followers may indeed announce that all prisoners have been interrogated before the leader does. However, for more than a few prisoners the likelihood of this is very low (see [24]).

5.1. A formalization of the puzzle

We first formalize the protocols in our framework. The leader is a prisoner that we name 0, and the followers are prisoners named $1, \ldots, n$ (with $n \geq 2$). The set $\mathbf{\Sigma}_{LB}$ of possible *actions* for the $n+1$ agents/prisoners $i = 0, \ldots, n$ is as follows:

name	description
t_i	i toggles
a_i	i announces
e_i	i enters
x_i	i exits

The set $\mathbf{P}_{LB}$ of relevant *atomic propositions* is as follows:

name	description
l	light is on
fin	protocol terminates
q_i	i has toggled the switch
m_i	the light *was* on, last time when i left the room (where $i \neq 0$)
p_0^j	0 has toggled the light for at least j times (where $0 \leq j \leq n$)
p_i^j	i has counted off-on changes for at least j times (where $i \neq 0$)

The post-conditions are given by the following table (where the remaining post-conditions are the identity).

(1)	$\iota(a_i, \mathit{fin})$	$=$	$\top$	$i \geq 0$
(2)	$\iota(x_i, m_i)$	$=$	l	$i \geq 0$
(3)	$\iota(t_i, q_i)$	$=$	$\top$	$i \geq 0$
(4)	$\iota(t_i, l)$	$=$	$\neg l$	$i \geq 0$
(5)	$\iota(t_0, p_0^j)$	$=$	$p_0^j \vee (p_0^{j-1} \wedge l)$	$j > 0$
(6)	$\iota(e_i, p_i^j)$	$=$	$p_i^j \vee (p_i^{j-1} \wedge ((\neg m_i \wedge l) \vee (\neg q_i \wedge \neg l)))$	$i > 0$

Post-condition (2) expresses that when i leaves the room he memorizes the situation of the light; post-condition (5) allows leader 0 to count the number of times that he toggled the switch; post-condition (6) lets i count the number of off-on changes. By Proposition 37, the above fact-changing actions $(\mathbf{\Sigma}_{LB}, \iota)$ can be turned into an equivalent factual change system $\mathcal{F}_{LB}$.

We are now ready to express the protocols in our protocol language.

Protocol 1 $\eta_1 = (?\neg fin \cdot \Sigma_{i=0}^{n}(e_i \cdot \theta_i \cdot x_i))^*$, where:

- $\theta_0 \;:= ?l \cdot t_0 \cdot (?p_0^n \cdot a_0 + ?\neg p_0^n) + ?\neg l$
- $\theta_i \;:= ?(\neg l \wedge \neg q_i) \cdot t_i + ?\neg(\neg l \wedge \neg q_i)$ $\qquad i > 0$

Protocol 2 $\eta_2 = (?\neg fin \cdot \Sigma_{i=0}^{n}(e_i \cdot \theta'_i \cdot x_i))^*$, where:

- $\theta'_0 \;:= \theta_0$
- $\theta'_i \;:= ?p_i^n \cdot a_i + ?\neg p_i^n \cdot \theta_i$ $\qquad i > 0$

It is not hard to see that the formulas θ_i are almost the literal translations of the specifications of Protocol 1. For $i > 0$, θ'_i only adds the extra announcement action based on θ_i. Note that θ_i, θ'_i are *deterministic* in the sense that there is always a unique way to proceed, due to the mutually exclusive preconditions of the actions.

5.2. Some example runs of the protocols

The initial situation can be represented as a singleton expectation model $\mathcal{M}, s$ with the universal protocol $\mathbf{\Sigma}_{LB}^*$ and the valuation assigning $\top$ *only* to p_i^0 for all $i \geq 0$.

Example 44. *Assume that there is a set of three prisoners* $\{0, 1, 2\}$ *and that the sequence of interrogations is* 1020*. We show an execution of Protocol 1 (formalized as* η_1*) on* $\mathcal{M}$*. Note that the followers do not need to count in Protocol 1, thus we omit all the* p_i^j, m_i *for* $i > 0$*:*

	l	fin	q_0	q_1	q_2	p_0^0	p_0^1	p_0^2
$\mathcal{M}$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$
e_1	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$
$t_1 \cdot x_1 \cdot e_0$	$\top$	$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\bot$
$t_0 \cdot x_0 \cdot e_2$	$\bot$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\bot$
$t_2 \cdot x_2 \cdot e_0$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\top$	$\top$	$\bot$
t_0	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$
a_0	$\bot$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$

In the above table, we combine several actions into a sequence if after the first action, the valuation of the relevant propositions stays the same throughout the whole sequence; for example, after t_1*, the valuation of the propositions in* *concern is not changed by x_1 and e_0. As the above table shows, 1 first turns the light on, 0 turns the light off, 2 turns the light on again, and finally 0 turns the light off and announces that everybody has been interrogated. Let $\eta_1^{\mathcal{F}_{LB}}$ be the singleton protocol model with respect to η_1 and $\mathcal{F}_{LB}$. Now we can verify the following:*

$$\mathcal{M}, s \vDash [!\eta_1^{\mathcal{F}_{LB}}]\langle e_1 \cdot t_1 \cdot x_1 \cdot e_0 \cdot t_0\rangle(\neg\langle a_0\rangle\top \wedge \langle x_0 \cdot e_2 \cdot t_2 \cdot x_2 \cdot e_0 \cdot t_0\rangle\langle a_0\rangle\top)$$

Formally, one needs first to convert η_1 with respect to $\mathcal{F}_{LB}$ into the corresponding normal form using the guarded automata construction of Proposition 41, and then construct the epistemic expectation model $\mathcal{M} \otimes \eta_1^{\mathcal{F}_{LB}}$ according to Definition 42, and finally check the truth value of the remaining $[!\eta_1^{\mathcal{F}_{LB}}]$-free formula on this model. For details of similar, rather involved, computations in the setting of other examples, see [4, p.47].

Example 45. *Still assuming that there are three prisoners, we now look at the interrogation sequence 1202 under Protocol 2. In the following table, the irrelevant propositions are omitted:*

	l	fin	q_0	q_1	q_2	p_0^0	p_0^1	p_0^2	m_2	p_2^0	p_2^1	p_2^2
$\mathcal{M}$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$
e_1	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$
$t_1 \cdot x_1$	$\top$	$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$
e_2	$\top$	$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\bot$
$x_2 \cdot e_0$	$\top$	$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\bot$
$t_0 \cdot x_0$	$\bot$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\bot$
e_2	$\bot$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\top$
a_2	$\bot$	$\top$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\top$

Follower 1 turns the light on; then follower 2 finds the light on and does not toggle the switch but counts 1; subsequently, leader 0 turns the light off; and finally follower 2 finds the light off, counts to 2 since he is ready to toggle the light, and then announces that everybody has been interrogated. Note that in the above table, m_2 plays an important role. We can verify:

$$\mathcal{M}, s \vDash [!\eta_2^{\mathcal{F}_{LB}}]\langle e_1 \cdot t_1 \cdot x_1 \cdot e_2 \cdot x_2 \cdot e_0 \cdot t_0\rangle(\neg\langle a_0\rangle\top \wedge \langle x_0 \cdot e_2\rangle\langle a_2\rangle\top)$$

5.3. Correctness of the two protocols

To check the correctness of the protocols, we need to show that if someone makes an announcement then each of the prisoners has been interrogated in the room at least once. Instead of this condition, we will actually check a stronger one, namely: If someone, say agent i, makes an announcement (a_i), then all the other prisoners $j \neq i$ have toggled the switch (q_j). Note that an agent can only make an announcement if he is in the room (e_i always precedes a_i in η_1 and η_2), thus it suffices to check q_j for all $j \neq i$. The correctness of Protocol 2 relies on the assumption that $n \geq 2$ ensures that the leader has toggled the light at least once before any follower can make the announcement. Formally, we can verify the following:

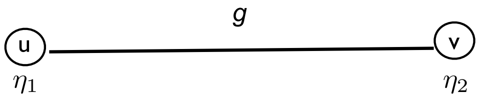
$$\mathcal{M}, s \vDash [!\eta_1^{\mathcal{F}_{LB}}][\mathbf{\Sigma}_{LB}^*](\langle a_0\rangle\top \to \bigwedge_{j\neq 0} q_i) \wedge [!\eta_2^{\mathcal{F}_{LB}}][\mathbf{\Sigma}_{LB}^*]\bigwedge_{i\geq 0}(\langle a_i\rangle\top \to \bigwedge_{j\neq i} q_j)$$

5.4. What does the guard know?

We can verify that the guard will always know when the prisoners will make the announcements, given that the protocol is public (recall that g is the guard). Let $\varphi_i = (\langle a_i\rangle\top \to K_g\langle a_i\rangle\top) \wedge (\neg\langle a_i\rangle\top \to K_g\neg\langle a_i\rangle\top)$. Now the following is straightforward, since there is only one world in the model throughout the evaluation:

$$\mathcal{M}, s \vDash [!\eta_1^{\mathcal{F}_{LB}}][\mathbf{\Sigma}_{LB}^*]\varphi_0 \wedge [!\eta_2^{\mathcal{F}_{LB}}][\mathbf{\Sigma}_{LB}^*]\bigwedge_{i\geq 0}\varphi_i$$

To confuse the guard, the prisoners may truthfully declare that they have agreed to use one of the two protocols, without telling the guard which one. Here we only model the uncertainty of the guard, not of the prisoners, by the following protocol model $\mathcal{A}^{\mathcal{F}_{LB}}$:



After updating $\mathcal{A}^{\mathcal{F}_{LB}}$ on $\mathcal{M}$, the new model $\mathcal{M}' = \mathcal{M} \otimes \mathcal{A}^{\mathcal{F}_{LB}}$ will have two g-indistinguishable states (s, u) and (s, v) with different expectations but the same valuation. For any $w \in \mathbf{\Sigma}_{LB}^*$, it is clear that the states in $\mathcal{M}'|_w$, if such states exist, have the same valuation, since the effect of executing w is deterministic. Therefore, the guard does not have any uncertainty about atomic propositions in $\mathbf{P}_{LB}$:

$$\mathcal{M}, s \vDash [!\mathcal{A}_v^{\mathcal{F}_{LB}}][\mathbf{\Sigma}_{LB}^*] \bigwedge_{p\in\mathbf{P}_{LB}} ((p \to K_g p) \wedge (\neg p \to K_g \neg p)).$$

On the other hand, an observation may be consistent with one state but not with the other. In particular, a sequence of actions ending by an announcement a_i may be possible on (s, v) but not possible on (s, u) since Protocol 2 (formalized as η_2) allows more prisoners to make the announcement, as was seen, for example, in the interrogation sequence 1202 in Example 45:

$$\mathcal{M}, s \not\models [!\mathcal{A}_v^{\mathcal{F}_{LB}}][\mathbf{\Sigma}_{LB}^*] \bigwedge_{0 \le i \le n} \varphi_i.$$

The above shows that the guard cannot always predict the announcements. On the other hand, he might find out which protocol the prisoners are running through his observations. The following formula says: If a follower does not announce that all prisoners have been interrogated in a situation in which he could do so according to Protocol 2, then the guard can eliminate the possibility that the prisoners are using Protocol 2 and make correct predictions of the future announcement:

$$\bigwedge_{0 < j \le n} [!\mathcal{A}_u^{\mathcal{F}_{LB}}][\mathbf{\Sigma}_{LB}^* \cdot e_j](p_j^n \to [(t_j + x_j) \cdot \mathbf{\Sigma}_{LB}^*] \bigwedge_{0 \le i \le n} \varphi_i).$$

Our language is very handy in verifying such complicated properties.

6. Related work

There are important differences between our work and the standard DEL approach with action models [13]. This summarizes those differences:

- In our setting the meaning of an action is not fixed. It is given by the expectations that come from protocols. For example, the way you interpret a fire depends on the protocol. It can be a warning or a welcome. There is no fixed precondition attached to the actions as in DEL.
- The π in the $[\pi]$ modalities in the language of POL are regular sets of action sequences. In DEL, in contrast, arbitrary finite action sequences (the Kleene * operator) are not commonly considered.
- Our protocol models look like action models in DEL but instead of preconditions we have protocols on each state, and the update with

such a model on an expectation model computes the expectations according to the protocols on each possible world of the expectation model, in contrast to the precondition matching in the standard DEL updates. Moreover, we introduce a notion of equivalence between protocol models based on the ideas of action emulation [20].

- Protocols in our setting are syntactic objects that are part of the logical language. In DEL, protocols are typically sets of sequences of DEL-actions.

We now continue with a more detailed comparison between our approach and DEL. In [4, 15], Wang introduces a logical framework for the dynamics of protocols and knowledge. In his framework, public protocols can be installed and changed, and the knowledge of agents is updated by matching expectations from protocols with observations. A similar update mechanism in the context of message passing can be found in the recent work [26] inspired by [2]. We also follow this type of 'matching updates' in this work, but deviate from [4, 15] by using epistemic models with explicit expectations, which we call epistemic expectation models, instead of standard epistemic models. Moreover, we use 'hidden protocols' on top of public ones.

Our epistemic expectation models may look similar to the models used in the work by Hoshi and colleagues [5, 3], where each epistemic state is equipped with an extensional DEL-protocol, namely a set of sequences of pointed action models. However, in the current article, a protocol is simply a syntactic expression based on tests and atomic actions that have neither inner structures nor fixed meanings. By using the protocol specification language, we can separate the protocols from epistemic models, and discuss the 'installation' of possibly uncertain protocol information on the epistemic models. In particular, we can formally discuss which kinds of expectations come from which kinds of protocols. Such a formal account of protocols also facilitates the study of the equivalence between protocols. We incorporate potentially iterative program-like observations, which also distinguishes us from the single-step updates in DEL-based protocol logics [26, 5, 3], where the iteration of updates often introduces undecidability, as observed in [27].

In [14], Pacuit and Simon present a PDL-style logic for reasoning about protocols under imperfect information. Their focus is on the executability and achievable outcomes of branching protocols under the uncertainties of the game states. In contrast, uncertainties may have *two* sources in the current paper: uncertainties about the real world and uncertainties about the protocols. The latter kind of uncertainty creates novel issues not covered by [14]. Executability of protocols also plays a role in our work but in a simpler way because of the linear interpretation of protocols, compared to the much more refined tree interpretation of protocols in [14]. Instead of executability, we focus more on the update effects of observations based on protocol information. In fact, the executors and the observers of the protocol can well be different. The protocol may be executed by external agents which are not modeled in the framework.

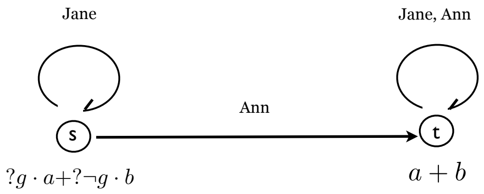
7. Conclusion and future work

The information that actions carry may depend on agents' knowledge of protocols. In this paper we studied cases where protocols are not commonly known and proposed a semantics-driven logical framework for updating knowledge by observations based on epistemic protocols. We have left a complexity analysis, for example, in line of [4], for the future. Although our semantics-driven logics POL and EPL are 'dynamic epistemic' in spirit, the usual reduction-based completeness proof for DEL-like logics does not apply, since the dynamic operators $[\pi]$ in POL cannot be eliminated. Complete axiomatizations of POL and EPL demand new techniques, pioneered in [28, 29]. We have partial results but we leave a systematic study to a future occasion. Let us consider various other extensions of our work.

We only used Boolean tests in the language $\mathcal{L}_{prot}$. A more expressive protocol language includes epistemic tests. An example of such a protocol would be $(?\neg Kp \cdot (a + b))^* \cdot (?Kp \cdot c)$: as long as you do not know p, keep choosing an a or b action, until you get to know p, and then do c. As observed in [30], knowledge-based protocols are much more involved than fact-based protocols. Defining the interpretation and executability of such protocols is a challenge, because checking epistemic formulas is nonlocal. Also, the introduction of knowledge tests may make the satisfiability problem of the logic undecidable. For example, the observations may easily encode iterated public announcement, which is known as a source of undecidability in such logics [27]. On the positive side, by including more expressive tests we expect better matching between epistemic expectation models and epistemic protocols (cf. Theorem 29).

Another extension is to consider less radical update mechanisms for installing new protocols. In our current approach, when installing a new protocol, we simply ignore and overwrite the old expected observations completely. Consider a singleton observation epistemic model with observation $a+c$. Now, when updating with the protocol $a+b$ we simply replace $a+c$ by $a+b$. Instead, we could integrate $a+c$ with $a+b$, somehow. For example, such a 'non-radical' protocol update with $a+b$ could result in b (intersected refinement), or in $(b+c)\cdot(a+b)$ (concatenation), or in $(b+c)+(a+b)$ (choice), and so on. See [15] for a discussion. Finally, we could relax the assumption of public observation, for example, some actions might not be observable to certain agents.

It would also be interesting to relax the underlying logic and to use KD45, modeling belief, instead of S5, modeling knowledge. For example, in the models of protocol updates for the story of Example 1 of the introduction (see page 18), it would fit more naturally with the story if the link for Ann between the alternatives in the epistemic protocol model were unidirectional only, namely from $?g\cdot a+?\neg g\cdot b$ to $a+b$, plus a Jane-loop from $?g\cdot a+?\neg g\cdot b$ to itself and Jane- and Ann-loops from $a+b$ to itself, as follows.



This would model installing the protocol wherein Anne is unaware of the gay interpretation.

Currently, the model on page 18 installs the possibly later observed information that Ann is uncertain whether the statement is to be interpreted as 'Kate is gay' or not, but she considers the option. By contrast, in the actual story, Jane will only interpret a as a sure sign of 'Kate is gay' and b as a sign of 'Kate is not gay'. We would rather be able to model that Jane considers both the 'Kate is gay' and the 'no double meaning' interpretation of a and b, corresponding to Ann's stance in the current model, whereas Ann *only* considers the 'no double meaning' interpretation and believes that Jane does so too.

The subject of hidden protocols is also interesting from the point of view of language pragmatics. Speakers who intend to convey information to only some of their listeners in such a way that others will not understand what is going on, are deliberately acting against some of Grice's maxims of cooperative conversation [31]. Forms of indirect or uncooperative communication, such as veiled bribes and threats, have already been investigated from the perspective of pragmatics and cognitive science, relating them also to aspects like lack of common knowledge [32, 33, 34, 35]. Our analysis of hidden protocols in this paper, by distinguishing between expected observations and actions, is more fine-grained than the changes in 'standard' dynamic epistemic logic, but could benefit from taking such Gricean aspects into account. Thus, in addition to observational powers of the agents, also their assertive powers may be modeled. Finally, it would be interesting to investigate the role of the interlocutors' goals and intentions when they utter a veiled speech act that is part of a hidden protocol (cf. [36, 37, 38, 39]).

Acknowledgments

Hans van Ditmarsch is also affiliated to IMSc (Institute of Mathematical Sciences, Chennai, India), as associated researcher. Part of this research was carried out by Hans van Ditmarsch and Yanjing Wang during their joint stays at the IMSc. Hans van Ditmarsch thanks the Netherlands Organization for Scientific Research (NWO) for a visiting grant 040.11.177 to Rineke Verbrugge, University of Groningen, of which the conference precursor [16] of this publication can be seen as an outcome. Hans van Ditmarsch also acknowledges support from ERC starting grant EPS 313360; this work has also been partially supported by the European Union Seventh Framework Programme under grant agreement no. 295261 (MEALS). Sujata Ghosh acknowledges NWO research grant 600.065.120.08N-201 and Rineke Verbrugge acknowledges NWO research grants 600.065.120.08N-201 and Vici grant NWO 227-80-001, both for her own research and for being able to invite Hans van Ditmarsch, Sujata Ghosh and Yanjing Wang as visitors to Groningen to cooperate on this extended article. Yanjing Wang thanks the National Social Science Foundation of China for the research grant 11CZX054 which also supports this work. He also acknowledges INSA-JRD TATA Fellowship which enabled him to visit Indian Statistical Institute (ISI), Chennai and helped in preparing the final version. The authors would like to thank the anonymous reviewers and the editors of this journal for their very helpful comments.

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

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

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

[4] Y. Wang, Epistemic Modelling and Protocol Dynamics, Ph.D. thesis, University of Amsterdam, 2010.

[5] T. Hoshi, Epistemic Dynamics and Protocol Information., Ph.D. thesis, Stanford University, 2009.

[6] Y. Zhang, Y. Zhou, Knowledge forgetting: Properties and applications, Artificial Intelligence 173 (2009) 1525–1537.

[7] F. Belardinelli, A. Lomuscio, Quantified epistemic logics for reasoning about knowledge in multi-agent systems, Artificial Intelligence 173 (2009) 982–1013.

[8] J. Halpern, Y. Moses, A guide to completeness and complexity for modal logics of knowledge and belief, Artificial Intelligence 54 (1992) 319–379.

[9] E. Davis, Knowledge and communication: A first-order theory, Artificial Intelligence 166 (2005) 81–139.

[10] S. Singh, The Code Book: The Evolution of Secrecy from Mary, Queen of Scots, to Quantum Cryptography, Doubleday, New York, NY, USA, 1999.

[11] A. van Kooten Niekerk, S. Wijmer, Verkeerde Vriendschap: Lesbisch Leven in de Jaren 1920-1960, Sara, Amsterdam, 1985.

[12] A. Baltag, A logic for suspicious players: Epistemic actions and belief-updates in games, Bulletin of Economic Research 54 (2002) 1–45.

[13] H. van Ditmarsch, W. van der Hoek, B. Kooi, Dynamic Epistemic Logic, volume 337 of *Synthese Library*, Springer, Berlin, 2007.

[14] E. Pacuit, S. Simon, Reasoning with protocols under imperfect information, The Review of Symbolic Logic 4 (2011) 412–444.

[15] Y. Wang, Reasoning about protocol change and knowledge, in: Proceedings of the 4th Indian Conference on Logic and its Applications (ICLA 2011), LNAI 6521, Springer, Berlin, 2010, pp. 189–203.

[16] H. van Ditmarsch, S. Ghosh, R. Verbrugge, Y. Wang, Hidden protocols, in: K. R. Apt (Ed.), Proceedings of the 13th Conference on Theoretical Aspects of Rationality and Knowledge (TARK-2011), ACM, 2011, pp. 65–74.

[17] J. A. Brzozowski, Derivatives of regular expressions, Journal of the ACM 11 (1964) 481–494.

[18] J. H. Conway, Regular Algebra and Finite Machines, Chapman and Hall, London, 1971.

[19] J. van Eijck, J. Ruan, T. Sadzik, Action emulation, Synthese 185 (2012) 131–151.

[20] D. van Eijck, F. Sietsma, Action emulation between canonical models, in: Proceedings of Conference on Logic and the Foundations of Game and Decision Theory 2012.

[21] J. van Benthem, J. van Eijck, B. Kooi, Logics of communication and change, Information and Computation 204 (2006) 1620–1662.

[22] J. van Eijck, Perception and change in update logic, in: J. van Eijck, R. Verbrugge (Eds.), Games, Actions and Social Software, volume 7010 of *Texts in Logic and Games (FOLLI subseries of LNCS)*, Springer Verlag, Berlin, 2011, pp. 119–140.

[23] D. Kozen, Automata on Guarded Strings and Applications, Technical Report, Cornell University, Ithaca, NY, USA, 2001.

[24] H. van Ditmarsch, J. van Eijck, W. Wu, Verifying one hundred prisoners and a lightbulb, Journal of Applied Non-Classical Logics 20 (2010) 173–191.

[25] H. van Ditmarsch, J. van Eijck, W. Wu, One hundred prisoners and a lightbulb - logic and computation, in: F. Lin, U. Sattler, M. Truszczynski (Eds.), KR, AAAI Press, 2010, pp. 90–100.

[26] B. Rodenhäuser, A logic for extensional protocols, Journal of Applied Non-Classical Logics 21 (2011) 477–502.

[27] J. S. Miller, L. S. Moss, The undecidability of iterated modal relativization, Studia Logica 79 (2005) 373–407.

[28] Y. Wang, Q. Cao, On axiomatizations of public announcement logic, Synthese (2013). Online first: http://dx.doi.org/10.1007/s11229-012-0233-5.

[29] Y. Wang, G. Aucher, An alternative axiomatization of DEL and its applications, in: Proceedings of IJCAI2013, pp. 1147–1154.

[30] R. Fagin, J. Y. Halpern, Y. Moses, M. Y. Vardi, Knowledge-based programs, Distributed Computing 10 (1997) 199–225.

[31] H. P. Grice, Logic and conversation, in: P. Cole, J. L. Morgan (Eds.), Syntax and Semantics, volume 3, New York: Academic Press, 1975, pp. 41–59.

[32] H. Clark, Using Language, Cambridge University Press, Cambridge, 1996.

[33] R. Verbrugge, L. Mol, Learning to apply theory of mind, Journal of Logic, Language and Information 17 (2008) 489–511. Special issue on formal models for real people, edited by M. Counihan.

[34] S. Pinker, M. Nowak, J. Lee, The logic of indirect speech, Bulletin of Economic Research 54 (2002) 1–45.

[35] H. van Ditmarsch, J. van Eijck, R. Verbrugge, Common knowledge and common belief, in: J. van Eijck, R. Verbrugge (Eds.), Discourses on Social Software, volume 5 of *Texts in Games and Logic*, Amsterdam University Press, Amsterdam, 2009, pp. 99–122.

[36] M. Bratman, Intention, Plans, and Practical Reason, Harvard University Press, Cambridge, MA, 1987.

[37] A. Rao, M. Georgeff, Modeling rational agents within a BDI-architecture, in: R. Fikes, E. Sandewall (Eds.), Proceedings of the Second Conference on Knowledge Representation and Reasoning, Morgan Kaufman, 1991, pp. 473–484.

[38] B. Grosz, C. Sidner, Plans for discourse, in: P. Cohen, J. Morgan, M. Pollack (Eds.), Intentions in Communication, MIT Press, Cambridge, MA, 1990, pp. 417–444.

[39] F. Dignum, B. Dunin-Kęplicz, R. Verbrugge, Creating collective intention through dialogue, Logic Journal of the IGPL 9 (2001) 145–158.