# Logic of Information Flow on Communication Channels*

Yanjing Wang[1], Floor Sietsma[2], and Jan van Eijck[2]

[1] Department of Philosophy, Peking University, Beijing 100871, China
[2] Centrum Wiskunde en Informatica,
P.O. Box 94079, NL-1090 GB Amsterdam, The Netherlands

**Abstract.** In this paper, we develop an epistemic logic for specifying and reasoning about information flow on the underlying communication channels. By combining ideas from Dynamic Epistemic Logic (DEL) and Interpreted Systems (IS), our semantics offers a natural and neat way of modeling multi-agent communication scenarios with different assumptions about the observational power of agents. We relate our logic to the standard DEL and IS approaches and demonstrate its use by studying a telephone call communication scenario.

## 1 Introduction

The 1999 'National Science Quiz' of *The Netherlands Organisation for Scientific Research (NWO)*[3] had the following question:

> *Six friends each have one piece of gossip. They start making phone calls. In every call they exchange all pieces of gossip that they know at that point. How many calls at least are needed to ensure that everyone knows all six pieces of gossip?*

To reason about the information flow in such a scenario, we want to take into account the following issues: the messages that the agents possess (e.g. secrets), the knowledge of the agents, the dynamics of the system in terms of information passing (e.g. telephone calls) and the underlying communication channels (e.g. the network of landlines). To incorporate specific designs for such issues, we first need to make a choice between two mainstream logical frameworks for multi-agent systems: *Interpreted Systems* and *Dynamic Epistemic Logic*.

*Interpreted Systems* (ISs), introduced by [15] and [8] independently, are mathematical structures that combine history-based temporal components of a system with epistemic ones (defined in terms of *local states* of the agents). ISs are convenient to model knowledge development based on the given temporal development

* This paper is the full version of an extended abstract with the same title appeared in the proceedings of AAMAS'10.

[3] For a list of references about the problem c.f. [12].

of a system. In ISs the epistemic structure is generated from the temporal structure in a uniform way. However, the generation of temporal structures is not specified in the framework.

A different perspective on the dynamics of multi-agent systems is provided by Dynamic Epistemic Logic (DEL) [9, 2]. The main focus of DEL is not on the temporal structure of the system but on the epistemic impact of events as the agents perceive them. The development of a system through time is essentially generated by executing so-called *action models* on a static initial model, to generate an updated static model. The epistemic relations in the initial static model and in the action models are not generated uniformly as in IS. Instead, they are designed by hand.

In recent years, much has been said about the comparison of the two frameworks, based on the observation that certain temporal developments of the system in IS can be generated by sequences of DEL updates on static models (see, e.g., [22, 11, 10]). In this paper, we will demonstrate further benefits of combining the two approaches by presenting a framework where epistemic relations are generated by matching local states and a history of observations as in ISs, while keeping the flexibility of explicit actions as in DEL approaches.

The puzzle of the telephone calls was briefly discussed in [25, Ch. 6.6] within the original DEL framework. In [21] the author raised the research question whether the communication network can be made explicit in DEL. An early proposal to fill in this line of research can be found in [19]. Communication channels in an IS framework made their appearance in [16]. Recent work in [14, 1] addresses the information passing on so-called *communication graphs* or *interaction structures*, where *"messages"* are either atomic propositions or Boolean combinations of atomic propositions. In [27] a PDL-style DEL language is developed that allows explicit specification of protocols. The present paper attempts to blend the DEL and IS approaches to model communication along channels. More specifically, the contributions of this paper are:

- Combining insights from DEL and IS, we propose a logic $\mathcal{L}^{I,M}$ to specify and reason about the information flow over underlying communication channels. Unlike the previous work [14, 1, 19], we can *specify* the communication protocols in our language and deal with information flow in terms of both *messages* and higher-order formulas.
- The semantics of $\mathcal{L}^{I,M}$ is given on single-state models with respect to different observational equivalence relations generated in IS-style, which are also studied and compared in this paper.
- The basic actions in $\mathcal{L}^{I,M}$ are given DEL-style internal structures by the semantics. This allows us to model various communicative actions such as message passing and group announcements. In particular we define an external informing action, which essentially announces the protocol that agents are supposed to follow, thus making it common knowledge. Therefore we can explicitly specify more details of epistemic protocols such as the ones discussed in [13]. It turns out to make a crucial difference whether epistemic

protocols are assumed to be common knowledge or not among the agents carrying out the protocol (see also [26, 27] for detailed discussions).
- Based on our semantics, we also propose a generic method of epistemic modeling where the initial model is simply the *real world* and all the initial assumptions are specified explicitly by means of formulas of $\mathcal{L}^{I,M}$. This significantly simplifies the modeling procedure. According to our semantics, the relevant possible states can be automatically constructed while evaluating the formulas. In particular, there is no need to specify the complete state space at the beginning.
- As a case study, we model telephone communications among agents. We show that it is impossible to obtain new common knowledge by telephone calls or voice mails but that we can get arbitrarily close to common knowledge if we not only can send messages but also make statements like "I know $j$ got message $m$".

The paper is organized as follows. We introduce our logic $\mathcal{L}^{I,M}$ in Section 2. Section 3 relates our logic to the standard DEL and IS approaches. Section 4 introduces a modeling method and illustrates this method by a study of variations on the puzzle that was mentioned above. The final section concludes and lists future work.

## 2 Logic $\mathcal{L}^{I,M}$

### 2.1 Language

Let $I$ be a finite set of agents, $M$ be a finite set of message terms, and $A$ be a finite set of basic actions. A communication network *net* is represented as a hypergraph of agents in $I$, namely a set of subsets of $I$ as in [1]. For example a hypergraph $net = \{\{1, 2\}, \{1, 2, 3\}\}$ denotes a network in which there is a private channel $\{1, 2\}$ between agents 1 and 2 and there is a public channel used by all three agents.

The set $Prop_{I,A,M}$ of basic propositions is defined by

$$p ::= has_i m \mid com(G) \mid past(\bar{\alpha}) \mid future(\bar{\alpha})$$

with $i \in I$, $m \in M$, $G \subseteq I$ and $\bar{\alpha} = \alpha_0; \alpha_1; \ldots; \alpha_k \in A^*$.

$has_i m$ is intended to mean that $i$ possesses the message $m$;[4] while $com(G)$ expresses that group $G$ forms a channel in the network; $past(\bar{\alpha})$ says that the sequence of actions $\bar{\alpha}$ just happened and $future(\bar{\alpha})$ means that $\bar{\alpha}$ can be executed according to the current protocol. The formulas of $\mathcal{L}^{I,M}$ are built from the set $Prop_{I,A,M}$ as follows:

$$\phi ::= \top \mid p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \langle\pi\rangle\phi \mid C_G\phi$$
$$\pi ::= \alpha \mid \varepsilon \mid \delta \mid \pi_1; \pi_2 \mid \pi_1 \cup \pi_2 \mid \pi^*$$

[4] *has* is a commonly used predicate in the logic of security protocols to model declarative knowledge about messages c.f., e.g., [18].

with $p \in Prop_{I,A,M}$, $G \subseteq I$, $\alpha \in A$ and $\varepsilon, \delta$ as constants for empty sequence and deadlock respectively.

The intended meaning of the formulas is mostly as usual as in dynamic epistemic logics: $C_G\phi$ expresses "*the agents in group $G$ commonly know $\phi$*", $\langle\pi\rangle\phi$ expresses "*the protocol $\pi$ can be executed, and at least one execution of $\pi$ yields a state where $\phi$ holds*".

As usual, we define $\bot$, $\phi \vee \psi$, $\phi \to \psi$, $\langle C_G\rangle\phi$ and $[\pi]\phi$ as the abbreviations of $\neg\top$, $\neg(\neg\phi \wedge \neg\psi)$, $\neg\phi \vee \psi$, $\neg C_G\neg\phi$ and $\neg\langle\pi\rangle\neg\phi$ respectively. Let $\Pi$ be the set of all protocols $\pi$. We also use the following additional abbreviations:

$$\begin{array}{|r:=l|}\hline K_j\phi & C_{\{j\}}\phi \\ has_i M' & \bigwedge_{m\in M'} has_i m \\ dhas_G M' & \bigwedge_{m\in M'} \bigvee_{j\in G} has_j m \\ com(net) & \bigwedge_{G\in net} com(G) \wedge \bigwedge_{G\notin net} \neg com(G) \\ \pi^n & \underbrace{\pi;\pi;\ldots;\pi}_{n} \\ \Sigma\Pi' & \bigcup_{\pi\in\Pi'} \pi \text{ where } \Pi' \subset \Pi \text{ is finite.} \\ \hline \end{array}$$

where $K_j\phi$ means that agent $j$ knows $\phi$; $dhas_G M'$ says the messages in $M'$ are distributed among agents in $G$; $com(net)$ specifies the communication channels in the network.

By having both *has* and $K$ operator in the language, we can make the distinction between knowing about a message and knowing about its content. $K_i has_j m \wedge \neg has_i m$ and $K_i has_j m \wedge has_i m$ can express the *de dicto* reading and *de re* reading of knowing a message $m$ respectively. For example, let $m$ be the hiding place of Bin Laden, then $K_{CIA} has_{Al-Qaeda} m \wedge \neg has_{CIA} m$ expresses that CIA knows that Al-Qaeda knows the hiding place, which is, however, a secret to CIA.

### 2.2 Semantics

First of all, we give interpretations to our action symbols $\alpha$ by defining their internal structures. Let $Form^{\text{-}\langle\pi\rangle}(\mathcal{L}^{I,M})$ be the set of all the $\mathcal{L}^{I,M}$ formulas without $\langle\pi\rangle$ modalities. Each $\alpha \in A$ can have an internal structure given by an *interpretation function* $\iota : A \to \mathcal{P}(I) \times Form^{-\langle\pi\rangle}(\mathcal{L}^{I,M}) \times (\mathcal{P}(M))^{|I|} \times (\Pi \cup \{\#\})$. Thus $\iota(\alpha)$ is a tuple:

$$\langle F, \phi, N_0 \ldots N_{|I|}, \rho\rangle$$

Here we define $Obs(\iota(\alpha)) = F$ as the set of agents that can observe $\alpha$; $Pre(\iota(\alpha)) = \phi$ is the precondition that should hold in order for $\alpha$ to be executable[5]; $Pos(\iota(\alpha)) = \langle N_0 \ldots N_{|I|}, \rho\rangle$ (with $\rho \in \Pi \cup \{\#\}$) is the postcondition which lists the set of messages $N_i$ that are delivered to $i$ by action $\alpha$ for each $i$ and the protocol $\rho$ that the agents are going to follow after executing $\alpha$. If $\rho = \#$, then the agents

[5] It will become clear when we define the semantics of $\mathcal{L}^{I,M}$ formulas that the action symbols in $\langle\pi\rangle$-free formulas are treated without referring to their internal structures given by $\iota$, thus avoiding the circularity in the definition of the semantics.

should keep following the current protocol. If $\rho = \pi$ for some $\pi \in \Pi$ then they should change their protocol to $\pi$. In this paper we assume that the agents can always observe the actions that deliver messages to them: if $N_j \neq \emptyset$ in $\iota(\alpha)$ then $j \in Obs(\iota(\alpha))$. The converse does not hold since agents may observe actions that do not deliver any messages to them.

Note that by excluding the preconditions in the form of $\langle\pi\rangle\phi$, the interdependence of actions are limited but still useful, e.g., for action $\alpha$, $future(\alpha)$ is allowed as a precondition meaning that $\alpha$ can be executed only when it was planned according to the current protocol.

In order to interpret basic propositions in $Prop_{I,A,M}$ we let the finer structure of the basic propositions correspond with a finer structure in the states, replacing the traditional valuation in Kripke structures used in the DEL-approaches:

**Definition 1.** *Let the state space* $S = \mathcal{P}(\mathcal{P}(I)) \times (\mathcal{P}(M))^{|I|} \times (A)^* \times (\mathcal{P}(M))^{|I|} \times \Pi$. *A state* $s \in S$ *for* $\mathcal{L}^{I,M}$ *is thus a tuple:*

$$\langle net, M_0, \ldots, M_{|I|}, \bar{\alpha}, M'_0, \ldots, M'_{|I|}, \pi \rangle$$

Here $IS(s,i) = M'_i$ is i's current set of messages (information set), $AM(s) = \bar{\alpha}$ is the action history, $CC(s) = net$ is the available communication network and $Prot(s) = \pi$ is the protocol that agents have to follow from this state. We let $AM_k(s) = \alpha_k$ in $\bar{\alpha}$ and $l(s) = |AM(s)|$ be the *length* of $s$. Note that each state also contains the information of the initial distribution of the messages: $M_0, \ldots, M_{|I|}$. From $s$ we can recover the initial state of the system before any actions were executed:

$$Init(s) = \langle net, M_0, \ldots, M_{|I|}, \epsilon, M_0, \ldots, M_{|I|}, (\Sigma A)^* \rangle.$$

The action history in the initial state is empty, thus $AM(Init(s)) = \epsilon$. We also assume that all the actions are allowed initially, thus $Prot(Init(s)) = (\Sigma A)^*$.

Intuitively, each state represents a past temporal development of the system with its constraint for the future actions. Note that the past is linear ($AM(s)$ is a single sequence of actions), while the future can be branching ($Prot(s)$ may allow several possible sequences of actions).

$has_i m$, $com(G)$ and $past(\bar{\alpha})$ can be interpreted in a straightforward way at a state $s$ according to $IS(s,i)$, $CC(s)$ and $AM(s)$ respectively. To give the semantics for $future(\bar{\alpha})$ at a state $s$, we need to check whether $\bar{\alpha}$ *complys* with the current protocol $Prot(s)$ and compute the remaining protocol after the execution of $\bar{\alpha}$ in order to know what the new protocol is. For this, we first recall the language of regular expressions $L(\pi)$:

$$\begin{array}{l} L(\delta) = \emptyset \qquad L(\varepsilon) = \{\epsilon\} \qquad L(\alpha) = \{\alpha\} \\ L(\pi;\pi') = \{\bar{\alpha};\bar{\beta} \mid \bar{\alpha} \in L(\pi), \bar{\beta} \in L(\pi')\} \\ L(\pi \cup \pi') = L(\pi) \cup L(\pi') \\ L(\pi^*) = \{\epsilon\} \cup \{\bar{\alpha}_1;\ldots;\bar{\alpha}_n \mid \bar{\alpha}_1, \ldots, \bar{\alpha}_n \in L(\pi)\} \end{array}$$

The language of an *input derivative* $\pi\backslash\bar{\alpha}$ of $\pi \in \Pi$ w.r.t. a sequence of actions $\bar{\alpha}$ is defined as $L(\pi\backslash\bar{\alpha}) = \{\bar{\beta} \mid \bar{\alpha};\bar{\beta} \in L(\pi)\}$(cf. [4]). Intuitively, $\pi\backslash\bar{\alpha}$ is the

remaining protocol of $\pi$ after executing $\bar{\alpha}$. The input derivatives can be computed efficiently e.g., we can derive $(\alpha \cup (\beta;\gamma))^* \backslash \beta = (\alpha \backslash \beta \cup (\beta;\gamma) \backslash \beta); (\alpha \cup \beta;\gamma)^* = (\delta \cup (\varepsilon;\gamma)); (\alpha \cup \beta;\gamma)^* = \gamma; (\alpha \cup (\beta;\gamma))^*$ (see [6] for an axiomatization of regular expression with input derivatives).

Similar to [5, 1], we give the truth value of complex $\mathcal{L}^{I,M}$ formula on *single* states instead of *pointed Kripke models*. The semantics of epistemic formulas depends on the action interpretation $\iota$ and the relation $\sim_i^x$ to be defined later. For any state $s$ we define:

$$\begin{array}{|rl|}\hline
s \vDash_\iota \mathit{has}_i(m) & \Leftrightarrow m \in \mathit{IS}(s,i) \\
s \vDash_\iota \mathit{com}(G) & \Leftrightarrow G \in \mathit{CC}(s) \\
s \vDash_\iota \mathit{past}(\bar{\alpha}) & \Leftrightarrow \bar{\alpha} \text{ is a suffix of } \mathit{AM}(s) \\
s \vDash_\iota \mathit{future}(\bar{\alpha}) & \Leftrightarrow \mathit{Prot}(s) \backslash \bar{\alpha} \neq \delta \\
s \vDash_\iota \neg\phi & \Leftrightarrow s \nvDash_\iota \phi \\
s \vDash_\iota \phi \wedge \psi & \Leftrightarrow s \vDash_\iota \phi \text{ and } s \vDash_\iota \psi \\
s \vDash_\iota C_G\phi & \Leftrightarrow \text{for all } v, \text{ if } s \sim_G^x t \text{ then } t \vDash_\iota \phi \\
s \vDash_\iota \langle\pi\rangle\phi & \Leftrightarrow \exists s' : s[\![\pi]\!]_\iota s' \text{ and } s' \vDash_\iota \phi \\
\hline
\end{array}$$

where $\sim_G^x$ is the reflexive transitive closure of $\bigcup_{i\in G} \sim_i^x$. The protocols $\pi$ function as *state changers* w.r.t. $\iota$:

$$\begin{array}{|rl|}\hline
s[\![\varepsilon]\!]_\iota s' & \Leftrightarrow s = s' \\
s[\![\delta]\!]_\iota s' & \Leftrightarrow \text{ never} \\
s[\![\alpha]\!]_\iota s' & \Leftrightarrow s \vDash_\iota \mathit{Pre}(\iota(\alpha)) \text{ and } s' = s|_{\mathit{Pos}(\iota(\alpha))} \\
s[\![\pi_1;\pi_2]\!]_\iota s' & \Leftrightarrow s[\![\pi_1]\!]_\iota \circ [\![\pi_2]\!]_\iota s' \\
s[\![\pi_1 \cup \pi_2]\!]_\iota s' & \Leftrightarrow s[\![\pi_1]\!]_\iota \cup [\![\pi_2]\!]_\iota s' \\
s[\![(\pi_1)^*]\!]_\iota s' & \Leftrightarrow s[\![\pi_1]\!]_\iota^* s' \\
\hline
\end{array}$$

where $\circ, \cup$ and $^*$ at right-hand side express the usual composition, union and reflexive transitive closure on relations respectively. Given $\mathit{Pos}(\iota(\alpha)) = \langle N_0, \ldots, N_{|I|}, \rho\rangle$, $s|_{\mathit{Pos}(\iota(\alpha))}$ is the result of executing action $\alpha$ at $s$ defined as:

$$s|_{\mathit{Pos}(\iota(\alpha))} = \langle \mathit{net}, M_0, \ldots, M_{|I|}, \bar{\beta}; \alpha, M_0' \cup N_0, \ldots, M_{|I|}' \cup N_{|I|}, f(\rho)\rangle$$

where $f(\rho) = \begin{cases} \pi \backslash \alpha & \text{if } \rho = \# \\ \pi' & \text{if } \rho = \pi' \end{cases}$.

Now we define $\sim_i^x$, the epistemic relation of an agent $i$ between states. A state $s$ is said to be *consistent* if $\mathit{Init}(s)[\![\mathit{AM}(s)]\!]_\iota s$. It is easy to see that for any $s$, $\mathit{Init}(s)$ is always consistent[6].

We define that $t \sim_i^x t'$ iff the following conditions are met:

**consistency** $t$ and $t'$ are consistent.
**local initialization** $\mathit{IS}(\mathit{Init}(t), i) = \mathit{IS}(\mathit{Init}(t'), i)$
**local history** $\mathit{AM}(t)|_i^x = \mathit{AM}(t')|_i^x$, where $x$ is the *type of observational power* of agents.

[6] Note that we can actually omit the current information sets $\mathit{IS}(s,i)$ in the definition of a state, and compute it by applying the actions in $\mathit{AM}(s)$, thus only generate consistent states. We keep the current information sets there to simplify notations and make it more efficient to evaluate basic propositions according to the semantics.

The type of observational power of the agents, $AM(t)|_i^x$, defines the local history. Many definitions of $AM(t)|_i^x$ are possible, giving the agents different observational powers. Several reasonable definitions are:

1. $AM(t)|_i^{set} = \{\alpha \text{ appearing in } AM(t) \mid i \in Obs(\iota(\alpha))\}$ as in [1].
2. $AM(t)|_i^{1st}$ is the subsequence of $AM(t)$ which only keeps the first occurrence of each $\alpha \in AM(t)|_i^{set}$ as in [3].
3. $AM(t)|_i^{asyn}$ is the subsequence of $AM(t)$ which only keeps all the occurrences of each $\alpha \in AM(t)|_i^{set}$, as in *asynchronous* systems (cf., e.g., [20]).
4. $AM(t)|_i^{\tau}$ is the sequence obtained by replacing each occurrence of $\alpha \notin AM(t)|_i^{set}$ in $AM(t)$ by $\tau$, as in *synchronous* systems with prefect recall (cf., e.g., [24]).

It is clear from the above definition that $\sim_i^x$ is an equivalence relation and the following holds:

**Proposition 1.** $\sim_i^{\tau} \subseteq \sim_i^{asyn} \subseteq \sim_i^{1st} \subseteq \sim_i^{set}$.

We then call the semantics defined by $\sim_i^x$ the *x-semantics*, and denote the corresponding satisfaction relation as $\models_\iota^x$. When $\iota$ is fixed and clear we also write $\models^x$ for the satisfaction relation. Recall that we require that the agents can always observe the actions that change his information set. Then we have:

**Proposition 2.** *For any consistent state $t, t'$: $t \sim_i^x t'$ implies $IS(t,i) = IS(t',i)$ where $x \in Sem = \{set, asyn, 1st, \tau\}$.*

*Proof.* By Proposition 1, $t \sim_i^x t'$ implies $t \sim_i^{set} t'$ for all $x \in Sem$. Therefore we only need to prove the claim for $x = set$. Suppose $t \sim_i^{set} t'$ then by the definition of $\sim_i^{set}$, $IS(Init(t), i) = IS(Init(t'), i)$ and $AM(t)|_i^{set} = AM(t')|_i^{set}$. So at $t$ and $t'$ agent $i$ initially had the same messages and has observed the same actions. Since agents can always observe the actions that change his information set then we know the same message passing actions relevant to $i$ have happened for $t$ and $t'$. Since the actions can only add messages to the information set and never delete messages from them, it doesn't matter how often or in which order those actions have been executed. Therefore the information sets of agent $i$ in $t$ and $t'$ are identical. □

### 2.3 Communicative Actions

In this section, we will define some useful basic actions with their internal structures[7]. To simplify the presentation, we abuse the notation of action names to stand for their internal structures as well, when the context is clear. Thus we let $Obs(\beta) = Obs(\iota(\beta))$ and similar for $Pre(\beta)$ and $Pos(\beta)$. Recall that the internal structure of an action $\beta$ is a tuple $\langle F, \phi, N_0, \ldots, N_{|I|}, \rho \rangle$ such that $N_j = \emptyset$ for $j \notin Obs(\beta)$. We now list some basic actions with their internal structures in Table 1.

[7] namely, a specific mapping $\iota$ which gives certain action names the corresponding internal structures

**Table 1.** Some important communicative actions

| $\beta$ (communication among the agents): | $Obs$ : | $Pre$ : common part is: $com(Obs(\beta)) \wedge future(\beta) \wedge$ | $Pos$ $(j \in Obs(\beta))$ : |
|---|---|---|---|
| $send^i_G(M')$ | $G \cup \{i\}$ | $has_i M'$ | $N_j = M', \rho = \#$ |
| $share_G(M')$ | $G$ | $dhas_G M'$ | $N_j = M', \rho = \#$ |
| $sendall^i_G(M')$ | $G \cup \{i\}$ | $has_i M' \wedge \bigwedge_{m \notin M'} \neg has_i m$ | $N_j = M', \rho = \#$ |
| $shareall_G(M')$ | $G$ | $dhas_G M' \wedge \bigwedge_{m \notin M'} \neg dhas_i m$ | $N_j = M', \rho = \#$ |
| $inform^i_G(\phi)$ | $G \cup \{i\}$ | $K_i \phi$ | $N_j = \emptyset, \rho = \#$ |
| $\beta$ (external actions): | $Obs$ : | $Pre$ : | $Pos$ : |
| $exinfo(\phi)$ | $I$ | $\phi$ | $\rho = \#$ |
| $exprot(\pi')$ | $I$ | $\top$ | $\rho = \pi'$ |

The first group of actions are communicative actions that are done by the agents. These actions must abide by the communication channels and the protocol, which is enforced by having $com(Obs(\beta)) \wedge future(\beta)$ in the precondition. $send^i_G(M')$ is the action that $i$ sends the set of messages $M'$ to the group $G$. Apart from respecting the channel and the protocol, the precondition $has_i M'$ enforces that agent $i$ should possess any messages he wants to send. The postcondition of $send^i_G(M')$ ensures that the messages in $M'$ are added to the message sets of the agents in $G$. The action $share_G(M')$ shares the messages in $M'$ within the group $G$. A precondition of $share_G(M')$ is that the messages from $M'$ are already distributed knowledge in the group. $sendall^i_G(M')$ differs from $send^i_G(M')$ in the extra precondition that $M'$ should contain *all* the messages that $i$ has. Similarly for $shareall_G(M')$. $inform^i_G(\phi)$ is the group announcement of an arbitrary formula $\phi$ within $G \cup \{i\}$. A precondition of $inform^i_G(\phi)$ is that $i$ should know $\phi$ is true before he can announce it.

The second group of actions are public announcements that do not respect the channels or the protocol. They model the external information that is given to the agents. $exinfo(\phi)$ models the public announcement of a formula $\phi$. The only precondition of this announcement is that $\phi$ should hold. The postcondition is empty. Knowledge of $\phi$ among the agents is created by the fact that all agents can observe the action. Since all agents know the execution of this action would only be possible if $\phi$ would hold, all agents know that $\phi$ holds at the moment it is announced. $exprot(\pi')$ announces the protocol $\pi'$ that the agents are supposed to follow in the future. Its postcondition changes the protocol to $\pi'$ and knowledge of the protocol is created because all agents observe the announcement.

We can define more complex actions based on the above basic actions, as we will demonstrate in Section 4.

## 3 Comparison with IS and DEL

The results in this section relate our logic to IS and DEL approaches. Theorem 1 shows that by the semantics of $\mathcal{L}^{I,M}$, an interpreted system is generated implic-

itly from a single state. Together with Theorem 1, Proposition 3 demonstrates that compared to DEL, our approach is powerful and concise in modeling actions. Let us compare our approach to IS first. In the following we only consider consistent states.

Let the history of $s$ w.r.t. a fixed $\iota$ be a sequence: $hist_\iota(s) = s_0 s_1 \dots s_{l(s)}$ where $s_0 = Init(s)$, $s_{l(s)} = s$ and $s_k [\![\alpha_k]\!]_\iota s_{k+1}$ for any $k$ such that $\alpha_k = AM_k(s)$. Clearly then $s_0 s_1 \dots s_k = hist_\iota(s_k)$ for any $k \leq l(s)$. Let $ExpT^x_\iota$ be the Interpreted System with action labels with respect to $x$-semantics $\{H, \{\rightarrow_\alpha |\ \alpha \in A\}, \{R_i \mid i \in I\}, V\}$, where:

- $H = \{hist_\iota(s) \mid s \text{ is consistent.}\}$
- $\langle s_0 \dots s_n\rangle \rightarrow_\alpha \langle s_0 \dots s_n s_{n+1}\rangle \Leftrightarrow s_n [\![\alpha]\!]_\iota s_{n+1}$.
- $\langle s_0 \dots s_n\rangle R_i \langle s'_0 \dots s'_m\rangle$ iff $s_n \sim^x_i s'_m$.
- $V(\langle s_0 \dots s_n\rangle)(p) = \top \Leftrightarrow s_n \models^x_\iota p$ where $p \in Prop_{I,A,M}$.

The language of $\mathcal{L}^{I,M}$ can be seen as a fragment of *Propositional Dynamic Logic* (PDL): $\mathcal{L}^I_{pdl}$ with basic action set $A \cup I$. Here the common knowledge operator $C_G$ can be seen as $[(\Sigma G)^*]$ in $\mathcal{L}^I_{pdl}$. Let $\Vdash_{PDL}$ denote the usual semantics of $\mathcal{L}^I_{pdl}$, then it is not hard to see:

**Theorem 1.** *For any formula $\phi \in \mathcal{L}^{I,M}$ and for each consistent $\mathcal{L}^{I,M}$-state $s$:*

$$s \models^x_\iota \phi \Leftrightarrow ExpT^x_\iota, hist_\iota(s) \Vdash_{PDL} \phi.$$

This result shows that if we abstract away the inner structure of basic propositions and actions, then our logic can be seen as a PDL language interpreted on ISs that are generated in a particular way w.r.t some constraints. Note that this result does not imply the decidability of $\mathcal{L}^{I,M}$ since although PDL is decidable on general Kripke structures, we do not know yet whether it is decidable on the restricted class of generated models $ExpT^x$.

Now consider the DEL language $\mathcal{L}^I_{del}$ :

$$\phi ::= \top \mid p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \langle \mathbb{A}, e\rangle\phi \mid C_G\phi$$

where $p$ is in a set of basic propositions $Prop$, $G \subseteq I$ and $\mathbb{A}$ is an *action model* with $e$ as its designated action. Action models are tuples of the form $(E, \{\succeq_i\}_{i\in I}, Pre, Pos)$ where $\succeq_i$ models agents $i$'s observational power on events in $E$ (e.g. $e_1 \succeq_i e_2$ means $i$ is not sure which one of $e_1$ and $e_2$ happened); the precondition function $Pre : E \rightarrow \mathcal{L}^I_{del}$ describes when an event can happen and the postcondition $Pos : E \rightarrow (Prop \rightarrow \mathcal{L}^I_{del})$ models the factual changes caused by the event by changing the truth value of basic proposition $p$ to the truth value of $Pos(e)(p)$ (cf. [2, 23] for details of action models). The semantics for epistemic formulas is as usual and

$$\mathbb{M}, s \Vdash_{DEL} \langle \mathbb{A}, e\rangle\phi \Leftrightarrow \mathbb{M} \otimes \mathbb{A}, (s, e) \models \phi$$

Where, given a static Kripke model $\mathbb{M} = (W, \{R_i\}_{i\in I}, V)$ and an action model $\mathbb{A} = (E, \{\succeq_i\}_{i\in I}, Pre, Pos)$, the updated model is $\mathbb{M} \otimes \mathbb{A} = (W', \{R'_i\}_{i\in I}, V')$ with:

$$W' = \{\langle w, e\rangle \mid \mathbb{M}, w \Vdash Pre(e)\}$$
$$R'_i = \{(\langle w, e\rangle, \langle v, e'\rangle) \mid wR_iv \text{ and } e \asymp_i e'\}$$
$$V'(\langle w, e\rangle)(p) = \top \Leftrightarrow \mathbb{M}, w \Vdash Pos(e)(p)$$

To facilitate a comparison, let us consider $\mathcal{L}^{I,M,-*}$, the star-free fragment of $\mathcal{L}^{I,M}$. Let $ExpK^x(s)$ be the Kripke model $\{W, \{R_i \mid i \in I\}, V\}$ obtained by the *expansion* of the state $s$ according to $x-$semantics, with:

- $W = \{s' \mid s \sim_I^x s'\}$ where $\sim_I^x$ is the reflexive transitive closure of $\{\sim_i^x \mid i \in I\}$.
- $R_i = \sim_i^x|_{W\times W}$.
- $V(s)(p) = \top \Leftrightarrow s \vDash^x p$ where $p \in Prop_{I,A,M}$.

Note that although $I, A, M$ are assumed to be finite, $W$ in $ExpK^x(s)$ can still be infinite due to the fact that we record the past explicitly in the states and there may be infinitely many possible histories.

Based on $ExpK^x(s)$ it seems plausible to obtain a similar correspondence result as Theorem 1 for $\mathcal{L}^{I,M,-*}$and $\mathcal{L}^I_{del}$, since the basic actions in $\mathcal{L}^{I,M,-*}$ look like special cases of pointed action models in DEL. However, the result does not hold in general. To see this, we first recall a fact from [22]: If we see $\langle \mathbb{A}, e\rangle$ as a basic action modality when considering $PDL$ semantics, then for any formula $\phi \in \mathcal{L}^I_{del}$ :

$$\mathbb{M}, s \Vdash_{DEL} \phi \Leftrightarrow Forest(\mathbb{M}, \mathcal{A}), (s) \Vdash_{PDL} \phi \quad (\star)$$

where $\mathcal{A}$ is the set of action models and $Forest(\mathbb{M}, \mathcal{A})$ is the IS generated by executing all possible sequences of action models in $\mathcal{A}$ on $\mathbb{M}, s$[8]. We now show that the effects of actions in $\mathcal{L}^{I,M}$ cannot be simulated by action models in general.

**Proposition 3.** *There exists some action interpretation $\iota$ such that there is no translation of action models $T : A \to \mathcal{A}$ satisfying:*

$$\textit{for all } \mathcal{L}^{I,M}\textit{-states } s\textit{: } T(ExpT^x_\iota), hist_\iota(s) \leftrightarrows Forest(ExpK^x(s), \mathcal{A}), (s)$$

*where $x \in \{set, 1st, asyn\}$ and $T(ExpT^x_\iota)$ is the IS obtained from $ExpT^x_\iota$ by replacing each label of $\alpha \in A$ by $T(\alpha) \in \mathcal{A}$ and $\leftrightarrows$ is the bisimulation w.r.t. transitions lablled by $I \cup \mathcal{A}$.*

*Proof.* In [22] it is shown that $Forest(ExpK^x(s), \mathcal{A})$ must satisfy the property of *Perfect Recall* meaning that if the agents can not distinguish two sequences of action $\bar{\alpha}; \alpha$ and $\bar{\beta}; \beta$ then they can not distinguish $\bar{\alpha}$ and $\bar{\beta}$. However, $ExpT^x$ clearly does not satisfy this property for $x \in \{set, 1st, asyn\}$ in general. For example, $send^i_j(M); \gamma \sim^x_j \gamma; send^i_j(M)$ where $x \in \{set, 1st, asyn\}$ and $\gamma$ is some action $j$ cannot observe, but $send^i_j(M) \not\sim^x_j \gamma$. □

If we consider $\tau-$semantics, then a correspondence result can be obtained. Given an action interpretation $\iota$, let $T^\iota_{DEL} : \mathcal{L}^{I,M,-*} \to \mathcal{L}^I_{del}$ be defined as follows:

[8] Due to the limit of space, readers are referred to [22] for details.

$$
\begin{array}{rcl}
T^{\iota}_{DEL}(\top) & = & \top \\
T^{\iota}_{DEL}(p) & = & p \\
T^{\iota}_{DEL}(\neg\phi) & = & \neg T^{\iota}_{DEL}(\phi) \\
T^{\iota}_{DEL}(\phi_1 \wedge \phi_2) & = & T^{\iota}_{DEL}(\phi_1) \wedge T^{\iota}_{DEL}(\phi_2) \\
T^{\iota}_{DEL}([\alpha]\phi) & = & [ExpA^{\tau}_{\iota}(\alpha)]T^{\iota}_{DEL}(\phi) \\
T^{\iota}_{DEL}([\pi_1 \cup \pi_2]\phi) & = & T^{\iota}_{DEL}([\pi_1]\phi) \wedge T^{\iota}_{DEL}([\pi_2]\phi) \\
T^{\iota}_{DEL}([\pi_1;\pi_2]\phi) & = & T^{\iota}_{DEL}([\pi_1][\pi_2]\phi)
\end{array}
$$

where $ExpA^{\tau}_{\iota}(\alpha)$ is the pointed action model $\{E, \{R_i \mid i \in I\}, V, e_\alpha\}$ obtained by the *saturation* of the action $\alpha$ according to $\tau-$semantics:

- $E = \{e_\beta \mid \beta \in A\}$
- $e_\beta R_i e_{\beta'} \Leftrightarrow \iota(\beta) = \iota(\beta')$ or $i \notin Obs(\beta) \cup Obs(\beta')$.
- $Pre(e_\beta) = T^{\iota}_{DEL}(Pre(\beta))$.
- If $Pos(\beta) = \langle N_0, \ldots, N_{|I|}, \rho\rangle$ then:

$$
Pos(e_\beta)(has_i m) = \begin{cases} \top & \text{if } m \in N_i \\ has_i m & \text{otherwise} \end{cases}
$$

$$
Pos(e_\beta)(com(G)) = com(G)
$$

$$
Pos(e_\beta)(past(\bar{\gamma};\gamma)) = \begin{cases} past(\bar{\gamma}) & \text{if } \gamma = \beta \\ \bot & \text{otherwise} \end{cases}
$$

$$
Pos(e_\beta)(future(\bar{\gamma})) = \begin{cases} future(\beta;\bar{\gamma}) & \text{if } \rho \text{ in } Pos(\beta) \text{ is } \sharp \\ \top & \text{if } \rho \text{ in } Pos(\beta) \text{ is } \pi \text{ and } \pi\backslash\bar{\gamma} \neq \delta \\ \bot & \text{if } \rho \text{ in } Pos(\beta) \text{ is } \pi \text{ and } \pi\backslash\bar{\gamma} = \delta \end{cases}
$$

Based on the above translation, the star-free fragment of $\mathcal{L}^{I,M}$ can be seen as a version of $DEL$ on generated models:

**Theorem 2.** *For any $\phi \in \mathcal{L}^{I,M,-*}$ and for any consistent $\mathcal{L}^{I,M}$-state $s$:*

$$
s \vDash^{\tau}_{\iota} \phi \Leftrightarrow ExpK^{\tau}_{\iota}(s), s \Vdash_{DEL} T_{DEL}(\phi).
$$

## 4 Applications

### 4.1 Common Knowledge

Our framework gives an interesting perspective on common knowledge. We first focus on asynchronous semantics. It may not be surprising that we cannot reach common knowledge without public communication. We might think that achieving common knowledge becomes easier if we can publicly agree on a common protocol before the communication is limited to non-public communication. However, in the case of asynchronous semantics we still can not reach common knowledge, even if we can publicly agree on a protocol. In this section we fix the action interpretation $\iota$ as in Section 2.3 thus omitting $\iota$ in $\vDash^{x}_{\iota}$ and $[\![\pi]\!]_\iota$. Recall that we say an action $\alpha$ *respects the communication channel* if $Pre(\alpha) \vDash com(Obs(\alpha))$.

**Theorem 3.** *For any state $s$ with $I \notin CC(s)$, any protocol $\pi$ containing only communications that respect the communication channels, any $\varphi \in \mathcal{L}^{I,M}$ and any sequence of actions $\bar{\alpha}$:*

$$s \vDash^{asyn} \langle exprot(\pi)\rangle(\neg C_I\varphi \to \neg\langle\bar{\alpha}\rangle C_I\varphi)$$

*Proof.* Let $s[\![exprot(\pi)]\!]t$ and suppose $t \vDash^{asyn} \neg C_I\phi$. Towards a contradiction, let $\bar{\alpha}$ be the minimal sequence of actions such that $t \vDash^{asyn} \langle\bar{\alpha}\rangle\phi$. Let $\bar{\alpha} = \bar{\beta};\alpha$, $t[\![\bar{\beta}]\!]u$ and $u[\![\alpha]\!]v$. Since $I \notin CC(s)$ and $\alpha$ respects the communication channel, $Obs(\alpha) \neq I$ so there exists $j \notin Obs(\alpha)$. Then $AM(u)|_j^{asyn} = AM(v)|_j^{asyn}$ so $u \sim_j^{asyn} v$. Since $\bar{\alpha}$ was minimal, $u \nvDash^{asyn} C_I\varphi$. But then $v \vDash^{asyn} \neg K_j C_I\varphi$, therefore $v \nvDash^{asyn} C_I\varphi$. □

Essentially, even if the agents agree on a protocol beforehand, the agents that cannot observe the final action of the protocol will never know whether this final action has been executed and thus common knowledge is never established. This is because in the asynchronous semantics, there is no sense of time. If we could add some kind of clock and the agents would agree to do an action on every "tick", the agents would be able to establish common knowledge. This is exactly what we try to achieve with our $\tau$-semantics. Here every agent observes a "tick" the moment some action is executed. This way, they can agree on a protocol *and* know when it is finished. We will show examples of how this can result in common knowledge in the next section on the telephone call scenario.

Here we will first investigate what happens in $\tau$-semantics if we *cannot* publicly agree on a protocol beforehand. We will show that in this case we cannot reach common knowledge of basic formulas. We start out with a lemma stating that actions preserve the agent's relations.

**Lemma 1.** *For any two states $s$ and $t$ and any action $\alpha$, if $s \sim_i^\tau t$ and we have $s'$, $t'$ such that $s[\![\alpha]\!]s'$ and $t[\![\alpha]\!]t'$ then $s' \sim_i^\tau t'$.*

*Proof.* Suppose $s \sim_i^\tau t$. Then $AM(s)|_i^\tau = AM(t)|_i^\tau$. Suppose $i \in Obs(\alpha)$. Then $AM(s')|_i^\tau = (AM(s)|_i^\tau;\alpha) = (AM(t)|_i^\tau;\alpha) = AM(t')|_i^\tau$. Suppose $i \notin Obs(\alpha)$. Then $AM(s')|_i^\tau = (AM(s)|_i^\tau;\tau) = (AM(t)|_i^\tau;\tau) = AM(t')|_i^\tau$. So $s' \sim_i^\tau t'$. □

This result may seem counter-intuitive, since for example a public announcement action may give the agents new information and thus destroy their epistemic relations. However, in our framework we model the new knowledge introduced by communicative actions by the fact that these actions would not be possible in states that do not satisfy the precondition of the action. In this lemma we assume that there are $s'$, $t'$ such that $s[\![\alpha]\!]s'$ and $t[\![\alpha]\!]t'$. This means that $s$ and $t$ both satisfy the preconditions of $\alpha$, so essentially no knowledge that distinguishes $s$ and $t$ is introduced by $\alpha$.

Now we define a fragment $\mathcal{L}_{bool}$ of our logic as follows:

$$\phi ::= has_i m \mid com(G) \mid \neg\phi \mid \phi_1 \wedge \phi_2$$

It is trivial to show that any action that does not change the agent's message sets or the protocol does not change the truth value of these basic formulas:

**Lemma 2.** *Let $\alpha$ be an action that can be executed on the state $s$ but does not change the agent's message sets or the protocol. For any $\phi \in \mathcal{L}_{bool}$: $s \vDash \phi \leftrightarrow \langle\alpha\rangle\phi$.*

Combining the properties of the actions from the previous lemma, we call an action $\alpha_d^G$ to be a *dummy action* for a group of agents $G$ if its internal structure has the precondition $com(G) \wedge future(\alpha_d^G)$, for it does not change the message sets of the agents or the protocol and $Obs(\alpha_d^G) = G$. An example of dummy action is $inform_G^i(\top)$. We could see it as "talking about irrelevant things".

**Theorem 4.** *Let $A$ be a set of basic actions respecting the communication channels such that for any agent $i$ there is a dummy action $\alpha_d^G$ such that $i \notin G$. Let $s$ be a state such that $I \notin CC(s)$ and it is common knowledge at $s$ that the protocol is $\pi = (\Sigma A)^*$ (any action in $A$ is allowed). Then for any $\phi \in \mathcal{L}_{bool}$ and any sequence of actions $\bar{\alpha}$,*

$$s \vDash^\tau \neg C_I\phi \rightarrow \neg\langle\bar{\alpha}\rangle C_I\phi$$

*Proof.* Similar to the proof of theorem 3, we suppose towards a contradiction that $s \vDash^\tau \neg C_I\phi$ and there is a minimal sequence $\bar{\alpha} = \bar{\beta};\alpha$ such that $s \vDash^\tau \langle\bar{\beta}\rangle(\neg C_I\phi \wedge \langle\alpha\rangle C_I\phi)$. Since $I \notin CC(s)$ and $\alpha$ respects the communication channel, there is $i \notin Obs(\alpha)$. Suppose $s[\![\bar{\beta}]\!]u$, then $u \vDash^\tau \neg C_I\phi$. Therefore there exists $u'$ such that $u \sim_I u'$ and $u' \vDash^\tau \neg\phi$. Now consider the dummy action $\alpha_d^G$ such that $i \notin G$. Clearly $\alpha_d^G$ can be executed on each state along the $\sim_I^\tau$ path from $u$ to $u'$. In particular there are $v$ and $v'$ such that $u[\![\alpha_d^G]\!]v$ and $u'[\![\alpha_d^G]\!]v'$. By lemma 1 it is not hard to see that $v \sim_I^\tau v'$. Since $\phi \in \mathcal{L}_{bool}$, by lemma 2 we have $v' \vDash^\tau \neg\phi$. Thus $v \nvDash^\tau C_I\phi$. Let $u[\![\alpha]\!]t$. Since $i \notin Obs(\alpha)$ and $i \notin Obs(\alpha_d^G)$, $AM(t)|_i^\tau = (AM(u)|_i^\tau;\tau) = AM(v)|_i^\tau$, thus $t \sim_i^\tau v$. Therefore $t \nvDash^\tau C_I\phi$, which contradicts our assumption.

□

### 4.2 Telephone Calls

Before going to the specific scenario of the telephone calls, we propose the following general modeling method:

1. Select a finite set of suitable actions $A$ with internal structures to model the communications in the scenario.
2. Design a single state as the *real world* to model the initial setting, i.e., $s = \langle net, \bar{M}_i, \epsilon, \bar{M}_i, (\Sigma A)^*\rangle$ where *net* models the communication network and $\bar{M}_i$ models *"who has what information"*.
3. Translate the informal assumptions of the scenario into formulas $\phi$ and protocols $\pi$ in $\mathcal{L}^{I,M}$.
4. Use $exinfo(\phi)$ and $exprot(\pi)$ to make the assumptions and the protocol common knowledge.

We will demonstrate how we use this method to model the telephone call scenario. Let us first recall the scenario: in a group of people, each person has one secret. They can make private telephone calls among themselves in order to

communicate these secrets. The original puzzle we mentioned in the introduction concerns the minimal number of telephone calls needed to ensure everyone gets to know all secrets. We start out by selecting a set of suitable actions $A$. We define:

$$\begin{aligned} call^i_j(M') &:= \bigcup_{M''\subseteq M'} shareall_{\{i,j\}}(M'') \\ mail^i_j(M') &:= \bigcup_{M''\subseteq M'} sendall^i_{\{j\}}(M'') \end{aligned}$$

Here $call^i_j(M')$ is the call between agent $i$ and $j$ where they share all messages out of $M'$ that they possess [9]. Later on we will also be interested in what happens if the agents can only leave *voicemail* messages instead of making two-way calls. For this purpose we use $mail^i_j(M')$, where agent $i$ sends all messages out of $M'$ he possesses to agent $j$. The third kind of communication we are interested in will be when the agents can call each other and communicate formulas instead of messages. This is modeled by $inform^i_j(\phi)$. Let $M_I = \{m_0, ..., m_{|I|}\}$ be the set of all secrets. For suitable finite sets of formulas $\Phi$ and protocols $\Pi$[10], we define

$$A = \bigcup_{\phi\in\Phi} exinfo(\phi) \cup \bigcup_{\pi\in\Pi} exprot(\pi) \cup \bigcup_{i,j\in I} call^i_j(M_I) \cup \bigcup_{i,j\in I} mail^i_j(M_I) \cup \bigcup_{i,j\in I,\phi\in\Phi} inform^i_j(\phi),$$

where we include $exinfo(\phi)$ and $exprot(\pi)$ because we need them to make the assumptions and the protocol of the scenario common knowledge.

Next, we define the communication network and the agent's message sets. Each agent has one secret so we define $M_i = \{m_i\}$. The agents can only communicate in pairs, so the communication network is $net^{tel}_I = \{\{i, j\} \mid i \neq j \in I\}$. Then the initial state is:

$$s^{tel}_I = \langle net^{tel}_I, \{m_0\} \dots \{m_{|I|}\}, \epsilon, \{m_0\} \dots \{m_{|I|}\}, (\Sigma A)^*\rangle$$

We are interested in situations with different communicative powers for the agents, which can be characterized by protocols that restrict the possible basic actions. We define $\pi_{call} := (\bigcup_{i,j\in I} call^i_j(M_I))^*$, $\pi_{mail} := (\bigcup_{i,j\in I} mail^i_j(M_I))^*$ as the protocols where the agents can only make telephone calls or send voicemails, respectively. We define $\pi_{\text{call, inform}} := (\bigcup_{i,j\in I} call^i_j(M_I) \cup \bigcup_{i,j\in I} mail^i_j(M_I))^*$.

As for the informal assumptions of the scenario, we assume it is common knowledge that every agent has one secret, and we assume the communication network is common knowledge. We use the following abbreviations:

$$\begin{aligned} \texttt{OneSecEach}_I &:= \textstyle\bigwedge_{i\in I}(has_i m_i \wedge \bigwedge_{j\neq i} \neg has_j m_i) \\ \texttt{TP} &:= exinfo(com(net^{tel}_I) \wedge \texttt{OneSecEach}_I) \\ \texttt{TP}_{act} &:= \texttt{TP}; exprot(\pi_{act}) \\ \texttt{HasAll}_I &:= \textstyle\bigwedge_{i\in I} has_i M_I \end{aligned}$$

$\texttt{OneSecEach}_I$ states that every agent has one secret known only to him. $\texttt{TP}_{act}$ is the action of announcing the assumptions of the scenario and protocol $\pi_{act}$ where $act \subseteq \{call, inform\}$. $\texttt{HasAll}_I$ expresses that every agent knows every secret, which is the goal we want to reach.

[9] Here $M'$ encodes the *relevant context* e.g. messages that are *"about work"*.

[10] For example, the sets of formulas/protocols up to the length of certain large number.

In order to reason about the number of calls the agents need to make to reach their goal, we use the following abbreviations:

$$\begin{aligned} \langle\rangle^{\leq n}\phi &:= \langle\bigcup_{k\leq n}(\Sigma A')^k\rangle\phi \\ \langle\rangle^{min(n)}\phi &:= \langle\rangle^{\leq n}\phi \wedge \neg\langle\rangle^{\leq n-1}\phi \end{aligned}$$

where $A'$ is the set of all actions in $A$ that respect the channels, i.e., excluding $exprot$, $exinfo$ and other external actions. $\langle\rangle^{\leq n}\phi$ expresses that we can reach a state where $\phi$ holds by sequentially executing at most $n$ actions from $A$ without external information or any changes in protocol. $\langle\rangle^{min(n)}\phi$ expresses that $n$ is the minimal such number. The reason we exclude these actions is because we essentially want to know whether we can reach $\phi$ with the current protocol. The external actions do not abide by the protocol, so we should not consider them[11].

Then the following result states that we need exactly $2|I|-4$ calls to make sure every agent knows all secrets:

**Proposition 4.** *For any $x \in Sem$:*

$$s_I^{tel} \models^x \langle \mathtt{TP}_{call}\rangle\langle\rangle^{min(2|I|-4)}\mathtt{HasAll}_I$$

A proof of this proposition is given in [12]. The protocol given there is the following: pick a group of four agents 1 ... 4 and let 4 be their *informant*. Let all other agents call agent 4, then let the four agents communicate all their secrets within their group and let all other agents call agent 4 again. In our framework we can express this as follows: $call_5^4(M_I); ...; call_{|I|}^4(M_I); call_2^1(M_I); call_4^3(M_I); call_3^1(M_I); call_4^2(M_I); call_5^4(M_I); ...; call_{|I|}^4(M_I)$.

Another interesting question arises when the agents cannot make direct telephone calls, but they can only leave voicemail messages. This means that any agent can tell the secrets he knows to another agent, but he cannot in the same call also learn the secrets the other agent knows. How many voicemail messages would we need in this case?

**Proposition 5.** *For any $x \in Sem$:*

$$s_I^{tel} \models^x \langle \mathtt{TP}_{mail}\rangle\langle\rangle^{min(2|I|-2)}\mathtt{HasAll}_I$$

*Proof.* Consider the following protocol: $mail_2^1(M_I); mail_3^2(M_I); ...; mail_{|I|}^{|I|-1}(M_I); mail_1^{|I|}(M_I); mail_2^{|I|}(M_I); ...; mail_{|I|-1}^{|I|}(M_I)$. Clearly, this results in all agents knowing all secrets. The length of this protocol is $2|I|-2$. We claim this protocol is minimal. To see why this claim holds, first observe that there has to be one agent who is the first to learn all secrets. For this agent to exist all other agents will first have to make at least one call to reveal their secret to someone else. This is already $|I|-1$ calls. The moment that agent learns all secrets, since he is the first, all other agents do not know all secrets. So each of them has to receive at least one more call in order to learn all secrets. This also takes $|I|-1$ calls which brings the total number of calls to $2|I|-2$. □

[11] Note that $\langle\rangle^{\leq n}$ serves as a generalization of the *arbitrary announcement* that is added to DEL in [17].

As we saw above, it is possible to make sure all agents know all secrets. However, in these results the secrets are not *common knowledge* yet, since the agents do not know that everyone knows all secrets. We will investigate whether we can establish common knowledge of $\mathtt{HasAll}_I$. If there are only three agents, this is possible by making telephone calls:

**Proposition 6.** *If $|I| \leq 3$ then for some $n \in \mathbb{N}$:*

$$s_I \vDash^\tau \langle \mathit{TP}_{call} \rangle \langle\rangle^{\leq n} C_I \mathtt{HasAll}_I$$

*Proof.* For $|I| < 3$ the proof is trivial. Suppose $|I| = 3$, say $I = \{1, 2, 3\}$. A protocol that results in the desired property is $\mathit{call}_2^1(M_I); \mathit{call}_3^2(M_I); \mathit{call}_1^2(M_I)$. After execution of this protocol all agents know all secrets. From the way they learned these secrets the agents can deduce what communications have happened. Since all agents can reason about each others knowledge it is common knowledge that all agents have all secrets. □

We do not extend this result for the case with more than three agents. If there are more than three agents, agents that are not participating in the phone call will never know which of the other agents are calling, which makes it much harder to establish common knowledge. A different interesting question is whether the agents will be able to reach common knowledge if they can tell each other arbitrary formulas of the language, using the *inform* action. This reduces the possibilities to reach common knowledge since the dummy action $\mathit{inform}_G^i(\top)$ is allowed. The agents have no clue whether any information is transferred when they observe a $\tau$ action so they can never reach common knowledge, not even in the case that $|I| = 3$. This directly follows from Theorem 4.

**Proposition 7.** *For any $n \in \mathbb{N}$, if $|I| > 2$ then:*

$$s_I \nvDash^\tau \langle \mathit{TP}_{call,inform} \rangle \langle\rangle^{\leq n} C_I \mathtt{HasAll}_I$$

Now imagine a situation where the agents are allowed to publicly announce beforehand a specific protocol they are going to follow which is more complex than just the set of actions they can choose from. Then, in our $\tau$-semantics, it is possible to reach common knowledge:

**Proposition 8.** *There is a protocol $\pi$ of call actions such that*

$$s_I \vDash^\tau \langle \mathit{TP}; \mathit{exprot}(\pi) \rangle \langle\rangle^{\leq n} C_I \mathtt{HasAll}_I$$

*Proof.* Let $\pi$ be the protocol given in the proof of proposition 4. Since each agent observes a $\tau$ at every communicative action, they can all count the number of communicative actions that have been executed and they all know when the protocol has been executed. So at that moment, it will be common knowledge that everyone has all secrets. □

This shows the use of the ability to communicate about the future protocol and not only about the past and present. There are many more situations where announcing the protocol is very important, for example in the puzzle of 100 prisoners and a light bulb [7] or many situations in distributed computing.

## 5 Conclusions and Future work

We developed an expressive dynamic epistemic logic tailored for specifying and reasoning about the information flow over communication channels. We also proposed an intuitive lightweight modeling method for multi-agent communication scenarios. The logic and the modeling method were put to use in the telephone call example.

Our framework is very flexible in modeling different observational powers of agents and various communicative actions. For example, we can define the communicative action in [14] : "$i$ gets $j$'s information without $j$ noticing that" as $\alpha = \mathit{download}^i_j(M)$ with $\mathit{Obs}(\iota(\alpha)) = i$, $\mathit{Pre}(\iota(\alpha)) = \mathit{com}(\{i,j\}) \wedge \mathit{has}_j M$ and a suitable postcondition adding messages to $i$'s information set[12]. Therefore our framework can facilitate the comparison among different approaches with different assumptions. The table below summarizes the setting of our framework compared to others:

| Reference | Actions | Information flow | Obs. Power |
|---|---|---|---|
| [19] | inform | propositions | $\equiv^{\tau}$ |
| [14] | download | Boolean atomic propositions | $\equiv^{\tau}$ |
| [1] | inform | positive atomic propositions | $\equiv^{set}$ |
| Our work | by design | messages or formulas | by design |

Among many others, we left the following issues for future work: the complexity analysis of the satisfiablity problem and model checking problem of $\mathcal{L}^{I,M}$; the more general communication channels e.g., asymmetric channels; actions that can change the communication channels (cf. [19]); other actions which are "partially observable" to agents, e.g., BCC in emailing; and announcements of protocols with tests (cf. [26] for further discussions).

## References


1. Krzysztof R. Apt, Andreas Witzel, and Jonathan A. Zvesper. Common knowledge in interaction structures. In Aviad Heifetz, editor, *Proceedings of TARK '09*, pages 4–13, 2009.
2. Alexandru Baltag and Lawrence Moss. Logics for epistemic programs. *Synthese*, 139(2):165–224, 2004.
3. Anguraj Baskar, Ramaswamy Ramanujam, and S. P. Suresh. Knowledge-based modelling of voting protocols. In Dov Samet, editor, *Proceedings of TARK '07*, pages 62–71, 2007.
4. Janusz A. Brzozowski. Derivatives of regular expressions. *Journal of the ACM*, 11(4):481–494, 1964.
5. Mika Cohen and Mads Dam. A complete axiomatization of knowledge and cryptography. In Luke Ong, editor, *Proceedings of LICS '07*, pages 77–88. IEEE Computer Society, 2007.
6. John H. Conway. *Regular Algebra and Finite Machines*. Chapman and Hall, 1971.
7. Paul O. Dehaye, Daniel Ford, and Henry Segerman. One hundred prisoners and a light bulb. *Mathematical Intelligencer*, 24(4):53–61, 2003.


[12] [14] phrases such download action with propositions instead of messages.

8. Ronald Fagin, Joseph Y. Halpern, Moshe Y. Vardi, and Yoram Moses. *Reasoning about Knowledge*. MIT Press, Cambridge, MA, USA, 1995.
9. Jelle Gerbrandy and Willem Groeneveld. Reasoning about information change. *Journal of Logic, Language and Information*, 6(2):147–169, 1997.
10. Tomohiro Hoshi. *Epistemic Dynamics and Protocol Information.* PhD thesis, Stanford University, 2009.
11. Tomohiro Hoshi and Audrey Yap. Dynamic epistemic logic with branching temporal structures. *Synthese*, 169(2):259–281, 2009.
12. Cor A. J. Hurkens. Spreading gossip efficiently. *Nieuw Archief voor Wiskunde*, 5/1(2):208–210, 2000.
13. Yoram Moses, Danny Dolev, and Joseph Y. Halpern. Cheating husbands and other stories: A case study of knowledge, action, and communication. *Distributed Computing*, 1(3):167–176, 1986.
14. Eric Pacuit and Rohit Parikh. Reasoning about communication graphs. In Johan van Benthem, Dov Gabbay, and Benedikt Löwe, editors, *Interactive Logic – Proceedings of the 7th Augustus de Morgan Workshop*, Texts in Logic and Games, pages 135–157, Amsterdam, 2007.
15. Rohit Parikh and Ramaswamy Ramanujam. Distributed processes and the logic of knowledge. In *Proceedings of the Conference on Logic of Programs*, pages 256–268, London, UK, 1985. Springer-Verlag.
16. Rohit Parikh and Ramaswamy Ramanujam. A knowledge based semantics of messages. *Journal of Logic, Language and Information*, 12(4):453–467, 2003.
17. Thomas Ågotnes, Philippe Balbiani, Hans van Ditmarsch, and Pablo Seban. Group announcement logic. *Journal of Applied Logic*, 8(1):62–81, 2009.
18. Ramaswamy Ramanujam and S. P. Suresh. Deciding knowledge properties of security protocols. In *Proceedings of TARK '05*, pages 219–235. Morgan Kaufmann, 2005.
19. Floris Roelofsen. Exploring logical perspectives on distributed information and its dynamics. Master's thesis, University of Amsterdam, 2005.
20. Nikolay V. Shilov and Natalya O. Garanina. Model checking knowledge and fixpoints. In Zoltán Ésik, Anna Ingólfsdóttir, Zoltán Ésik, and Anna Ingólfsdóttir, editors, *Proceedings of FICS '02*, volume NS-02-2, pages 25–39, 2002.
21. Johan van Benthem. 'One is a lonely number': on the logic of communication. In Z. Chatzidakis, P. Koepke, and W. Pohlers, editors, *Proceedings of Logic Colloquium '02*, pages 96–129, Wellesley MA, 2002. ASL & A.K. Peters.
22. Johan van Benthem, Jelle Gerbrandy, Tomohiro Hoshi, and Eric Pacuit. Merging frameworks for interaction. *Journal of Philosophical Logic*, 38(5):491–526, October 2009.
23. Johan van Benthem, Jan van Eijck, and Barteld Kooi. Logics of communication and change. *Information and Computation*, 204(11):1620–1662, 2006.
24. Ron van der Meyden and Nikolay Shilov. Model checking knowledge and time in systems with perfect recall. In *Proceedings of FSTTCS '99*, pages 432–445, 1999.
25. Hans van Ditmarsch. *Knowledge Games.* PhD thesis, University of Groningen, 2000.
26. Yanjing Wang. *Epistemic Modelling and Protocol Dynamics.* PhD thesis, University of Amsterdam, 2010.
27. Yanjing Wang, Lakshmanan Kuppusamy, and Jan van Eijck. Verifying epistemic protocols under common knowledge. In Aviad Heifetz, editor, *Proceedings of TARK '09*, pages 257–266, 2009.