
Logic of Information Flow on Communication Channels

Yanjing Wang*, Floor Sietsma and Jan van Eijck

Centrum Wiskunde en Informatica,

y.wang, f.sietsma, Jan.van.Eijck@cwi.nl

Abstract

In this paper¹, we develop an epistemic logic to specify and reason 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 modelling 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)*² 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.

*The first author is supported by NWO project VEMPS 612.000.528.

¹This paper is based on an extended abstract to appear in the proceedings of AAMAS10.

²For a list of references about the problem c.f. Hurkens (2000).

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 Parikh and Ramanujam (1985) and Fagin et al. (1995) 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 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) (Gerbrandy and Groeneveld 1997, Baltag and Moss 2004). 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. It is customary to start out from a static situation of universal ignorance, where the ignorance is supposed to be common knowledge³.

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., van Benthem et al. (2009), Hoshi and Yap (2009), Hoshi (2009)). 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.

³In a situation with n atomic propositions, this gives an initial model consisting of 2^n worlds, with universal accessibility relations for all agents.

The puzzle of the telephone calls was briefly discussed in van Ditmarsch (2000, Ch. 6.6) within the original DEL framework. van Benthem (2002) 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 Roelofsen (2005). Communication channels in an IS framework made their appearance in Parikh and Ramanujam (2003). Recent work in (Pacuit and Parikh 2007, Apt et al. 2009) 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 Wang et al. (2009) 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 Dynamic Epistemic Logics and Interpreted Systems, we propose a logic $\mathcal{L}_i^{I,M}$ to specify and reason about the information flow over underlying communication channels. Unlike in previous work in Pacuit and Parikh (2007), Apt et al. (2009), Roelofsen (2005), 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 $\mathcal{L}_i^{I,M}$ 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 DEL-style actions in $\mathcal{L}_i^{I,M}$ allow 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 that the future behavior of the agents is constrained. It turns out to make a crucial difference whether epistemic protocols such as those discussed in van Ditmarsch et al. (2007) are assumed to be common knowledge among the agents carrying out the protocol or not (see also Wang et al. (2009)).
 - Taking advantage of 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 $\mathcal{L}_i^{I,M}$. 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 whole 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 $\mathcal{L}_t^{I,M}$ 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 $\mathcal{L}_t^{I,M}$

2.1 Language

Let I be a finite set of agents, M a finite set of message terms and A a finite set of basic actions with internal structures given by an action map ι defined later. A communication network net is represented as a hypergraph of agents in I , namely a set of subsets of I as in Apt et al. (2009). For example if $net = \{\{1, 2\}, \{1, 2, 3\}\}$ then 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 $\bar{\alpha} = \alpha_0; \alpha_1; \dots; \alpha_k \in A^*$.

$has_i m$ is intended to mean that i possesses the message m^A ; 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 $\mathcal{L}_t^{I,M}$ are built from the set $Prop_{I,A,M}$ as follows:

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

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

⁴*has* is a commonly used predicate in the logic of security protocols to model declarative knowledge about messages c.f., e.g., Ramanujam and Suresh (2005).

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 ϕ ”, $\langle\pi\rangle\phi$ expresses “the protocol π can be executed, and at least one execution of π yields a state where ϕ holds”. Let Π be the set of all protocols π and $Form^{-\langle\pi\rangle}(\mathcal{L}_i^{I,M})$ be the set of all the $\mathcal{L}_i^{I,M}$ formulas without $\langle\pi\rangle$ modalities. Each $\alpha \in A$ has an internal structure given by $\iota : A \rightarrow \mathcal{P}(I) \times Form^{-\langle\pi\rangle}(\mathcal{L}_i^{I,M}) \times (\mathcal{P}(M))^{|I|} \times (\Pi \cup \{\#\})$. Thus $\iota(\alpha)$ is a tuple:

$$\langle G, \phi, M_0 \dots M_{|I|}, \rho \rangle$$

Here we define $Obs(\iota(\alpha)) = G$ as the set of agents that can observe α ; $Pre(\iota(\alpha)) = \phi$ is the precondition that should hold in order for α to be executable; $Pos(\iota(\alpha)) = \langle M_0 \dots M_{|I|}, \rho \rangle$ (with $\rho \in \Pi \cup \{\#\}$) is the postcondition which lists the set of messages M_i that get delivered to i by action α for each i and the protocol ρ that the agents are going to follow after execution of α . If $\rho = \#$, then the agents should keep following the current protocol. If $\rho = \pi$ for some $\pi \in \Pi$ then they should change their protocol to π . In this paper we assume that the agents can always observe the actions that deliver messages to them: if $\iota(\alpha)$ has $M_j \neq \emptyset$ then $j \in Obs(\iota(\alpha))$. The converse does not hold since agents may also observe actions that do not deliver any messages to them.

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

As usual, we define \perp , $\phi \vee \psi$, $\phi \rightarrow \psi$, $\langle C_G \rangle \phi$ 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. Moreover, we use the following additional abbreviations:

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

where $K_j\phi$ means that agent j knows ϕ ; $dhas_G M'$ says the messages from M' are distributed knowledge 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* 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

In order to interpret basic propositions $Prop_{I,A,M}$ 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 DEL-approaches:

Definition 2.1. 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 $\mathcal{L}_i^{I,M}$ is thus a tuple:

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

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 information of the initial distribution of the messages: $M_0, \dots, M_{|I|}$. From s we can recover the initial state of the system before any actions were executed:

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

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 recall the *input derivate* $\pi \setminus \alpha$ of the regular expression $\pi \in \Pi$ and the output function $o : \Pi \rightarrow \{\delta, \epsilon\}$ (cf.

Brzozowski (1964), Conway (1971)):

$$\begin{array}{ll}
\varepsilon \backslash \alpha = \delta \backslash \alpha = \beta \backslash \alpha = \delta & (\alpha \neq \beta) & \alpha \backslash \alpha = \varepsilon \\
(\pi; \pi') \backslash \alpha = (\pi \backslash \alpha); \pi' \cup o(\pi); (\pi' \backslash \alpha) & & (\pi \cup \pi') \backslash \alpha = \pi \backslash \alpha \cup \pi' \backslash \alpha \\
(\pi^*) \backslash \alpha = \pi \backslash \alpha; (\pi^*)^* & & o(\pi; \pi) = o(\pi); o(\pi') \\
o(\pi^*) = \varepsilon & & o(\varepsilon) = \varepsilon \\
o(\delta) = o(\alpha) = \delta & & o(\pi \cup \pi') = o(\pi) \cup o(\pi')
\end{array}$$

Let $\pi \backslash (\alpha_0; \alpha_1; \dots; \alpha_n) = (\pi \backslash \alpha_0) \backslash \alpha_1 \dots \backslash \alpha_n$. Together with the axioms of Kleene algebra we can derive syntactically the remaining protocol after executing a sequence of basic actions. For example: $(\alpha \cup (\beta; \gamma))^* \backslash \beta = (\alpha \backslash \beta \cup (\beta; \gamma) \backslash \beta); (\alpha \cup \beta; \gamma)^* = (\delta \cup (\varepsilon; \gamma)); (\alpha \cup \beta; \gamma)^* = \gamma; (\alpha \cup (\beta; \gamma))^*$. Note that in general we do not have $\bar{\beta}; (\pi \backslash \bar{\beta}) = \pi$.

Let $L(\pi)$ be the language of the regular expressions defined as follows:

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

From Conway (1971), we have:

Proposition 1. $L(\pi \backslash \bar{\alpha}) = \{\bar{\beta} \mid \bar{\alpha}; \bar{\beta} \in L(\pi)\}$.

Similar to Cohen and Dam (2007), Apt et al. (2009), we give the truth value of complex $\mathcal{L}_i^{I,M}$ formula on *single* states instead of *pointed Kripke models*. The interpretation of epistemic formulas depends on the relation \sim_i^x to be defined later.

For any state s we define:

$s \models has_i(m)$	\Leftrightarrow	$m \in IS(s, i)$
$s \models com(G)$	\Leftrightarrow	$G \in CC(s)$
$s \models past(\bar{\alpha})$	\Leftrightarrow	$\bar{\alpha}$ is a suffix of $AM(s)$
$s \models future(\bar{\alpha})$	\Leftrightarrow	$Prot(s) \backslash \bar{\alpha} \neq \delta$
$s \models \neg \phi$	\Leftrightarrow	$s \not\models \phi$
$s \models \phi \wedge \psi$	\Leftrightarrow	$s \models \phi$ and $s \models \psi$
$s \models C_G \phi$	\Leftrightarrow	for all v , if $s \sim_G^x t$ then $t \models \phi$
$s \models \langle \pi \rangle \phi$	\Leftrightarrow	$\exists s' : s \ll[\pi] s'$ and $s' \models \phi$

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

$s \llbracket \varepsilon \rrbracket s'$	\Leftrightarrow	$s = s'$
$s \llbracket \delta \rrbracket s'$	\Leftrightarrow	never
$s \llbracket \alpha \rrbracket s'$	\Leftrightarrow	$s \vDash Pre(t(\alpha))$ and $s' = s _{Pos(t(\alpha))}$
$s \llbracket \pi_1; \pi_2 \rrbracket s'$	\Leftrightarrow	$s \llbracket \pi_1 \rrbracket \circ \llbracket \pi_2 \rrbracket s'$
$s \llbracket \pi_1 \cup \pi_2 \rrbracket s'$	\Leftrightarrow	$s \llbracket \pi_1 \rrbracket \cup \llbracket \pi_2 \rrbracket s'$
$s \llbracket (\pi_1)^* \rrbracket s'$	\Leftrightarrow	$s \llbracket \pi_1 \rrbracket^* s'$

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

$$s|_{Pos(t(\alpha))} = \langle net, M_0, \dots, M_{|I|}, \bar{\beta}; \alpha, M'_0 \cup N_0, \dots, M'_{|I|} \cup N_{|I|}, f(\rho) \rangle$$

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

Now we define \sim_i^x , the epistemic relation of an agent i between states. A state s is said to be *consistent* if $Init(s) \llbracket AM(s) \rrbracket s$. It is easy to see that for any s , $Init(s)$ is always consistent⁵.

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

consistency t and t' are consistent.

local initialization $IS(Init(t), i) = IS(Init(t'), i)$

local history $AM(t)_i^x = AM(t')_i^x$, where x is the *type of observational power* of agents.

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. $AM(t)_i^{set} = \{\alpha \text{ appearing in } AM(t) \mid i \in Obs(t(\alpha))\}$ as in Apt et al. (2009).
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 Baskar et al. (2007).
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., Shilov and Garanina (2002)).

⁵Note that we can actually omit the current information sets $IS(s, i)$ in the definition of a state, and compute it by applying the actions in $AM(s)$, 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.

4. $AM(t)|_i^\tau$ is the sequence obtained by replacing each occurrence of $\alpha \notin AM(t)|_i^{set}$ in $AM(t)$ by τ , as in *synchronous* systems with perfect recall (cf., e.g., van der Meyden and Shilov (1999)).

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

Proposition 2. $\sim_i^\tau \subseteq \sim_i^{asyn} \subseteq \sim_i^{1st} \subseteq \sim_i^{set}$.

We then call the semantics defined by \sim_i^x the *x-semantics*, and denote the corresponding satisfaction relation as \models^x .

Recall that we require that the agents can always observe the actions that change his information set. Then we have:

Proposition 3. For any consistent state t : $t \sim_i^x t'$ implies $IS(t, i) = IS(t', i)$ where $x \in Sem = \{set, asyn, 1st, \tau\}$.

Proof. By Proposition 2, $t \sim_i^x t'$ 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. \square

2.3 Communicative Actions

In this section, we will define some useful basic actions with their internal structures. 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 β is a tuple $\langle F, \phi, N_0, \dots, N_{||}, \rho \rangle$ such that $N_j = \emptyset$ for $j \notin Obs(\beta)$. We now list some basic actions with their internal structures defined in the table below:

β (communication by the agents):	<i>Obs</i> :	<i>Pre</i> : common part is: $com(Obs(\beta)) \wedge future(\beta) \wedge \dots$	<i>Pos</i> $(j \in Obs(\beta)) :$
$send_G^i(M')$	$G \cup \{i\}$	$has_i M'$	$N_j = M', \rho = \#$
$share_G(M')$	G	$dhas_G M'$	$N_j = M', \rho = \#$
$sendall_G^i(M')$	$G \cup \{i\}$	$has_i M' \wedge \bigwedge_{m \in M'} \neg has_i m$	$N_j = M', \rho = \#$
$shareall_G(M')$	G	$dhas_G M' \wedge \bigwedge_{m \in M'} \neg dhas_i m$	$N_j = M', \rho = \#$
$inform_G^i(\phi)$	$G \cup \{i\}$	$K_i \phi$	$N_j = \emptyset, \rho = \#$
β (external actions):	<i>Obs</i> :	<i>Pre</i> :	<i>Pos</i> :
$exinfo(\phi)$	I	ϕ	$\rho = \#$
$exprot(\pi')$	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. $send_G^i(M')$ 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 $send_G^i(M')$ expresses that the messages in M' get added to the message sets of the agents in G . $share_G(M')$ shares the messages from M' within the group G . A precondition is that the messages from M' are already distributed knowledge in the group. $sendall_G^i(M')$ differs from $send_G^i(M')$ in the extra precondition that M' should contain *all* the messages that i has. Similarly for $shareall_G(M')$. $inform_G^i(\phi)$ is the group announcement of an arbitrary formula ϕ within $G \cup \{i\}$. A precondition is that i should know ϕ 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 ϕ . The only precondition of this announcement is that ϕ should hold. The postcondition is empty. Knowledge of ϕ 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 ϕ would hold, all agents know that ϕ holds at the moment it is announced. $exprot(\pi')$ announces the protocol π' that the agents are supposed to follow in the future. Its postcondition changes the protocol to π' 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 $\mathcal{L}_i^{I,M}$, an interpreted system is generated implicitly from a single state. Together with Theorem 1, Proposition 4 demonstrates that compared to DEL, our approach is powerful and concise in modelling actions. Let us compare our approach to IS first. In the following we only consider consistent states.

Let the history of s be a sequence: $hist(s) = s_0s_1 \dots s_{l(s)}$ where $s_0 = Init(s)$, $s_{l(s)} = s$ and $s_k \llbracket \alpha_k \rrbracket s_{k+1}$ for any k such that $\alpha_k = AM_k(s)$. Clearly then $s_0s_1 \dots s_k = hist(s_k)$ for any $k \leq l(s)$. Let $ExpT^x$ be the Interpreted System with action labels with respect to x -semantics $\{H, \rightarrow_\alpha, \{R_i \mid i \in I\}, V\}$, where:

- $H = \{hist(s) \mid s \text{ is consistent.}\}$
- $\langle s_0 \dots s_n \rangle \rightarrow_\alpha \langle s_0 \dots s_n s_{n+1} \rangle \Leftrightarrow s_n \llbracket \alpha \rrbracket s_{n+1}$.
- $\langle s_0 \dots s_n \rangle R_i \langle s'_0 \dots s'_m \rangle$ iff $s_n \sim_i^x s'_m$.
- $V(\langle s_0 \dots s_n \rangle)(p) = \top \Leftrightarrow s_n \models^x p$ where $p \in Prop_{I,A,M}$.

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

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

$$s \models^x \phi \Leftrightarrow ExpT^x, hist(s) \Vdash_{PDL} \phi.$$

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 $\mathcal{L}_i^{I,M}$ 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}_{del}^I :

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

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 $(E, \{\simeq_i\}_{i \in I}, Pre, Pos)$ where \simeq_i models agents i 's observational power on events in E (e.g. $e_1 \simeq_i e_2$

means i is not sure which one of e_1 and e_2 happened); the precondition function $Pre : E \rightarrow \mathcal{L}_{del}^I$ describes when an event can happen and the postcondition $Pos : E \rightarrow (Prop \rightarrow \mathcal{L}_{del}^I)$ models the factual changes caused by the event by changing the truth values of basic propositions p to $Pos(e)(p)$ van Benthem et al. (2006). 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 $\mathbb{A} = (E, \{\asymp_i\}_{i \in I}, Pre, Pos)$, the updated model is $\mathbb{M} \otimes \mathbb{A} = (W', \{R'_i\}_{i \in I}, V')$ with:

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

To facilitate a comparison, let us consider $\mathcal{L}_t^{I,M,-*}$, the star-free fragment of $\mathcal{L}_t^{I,M}$. 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 $\{\sim_i^x \mid i \in I\}$.
- $R_i = \sim_i^x|_{W \times W}$.
- $V(s)(p) = \top \Leftrightarrow s \models^x p$ 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 $\mathcal{L}_t^{I,M,-*}$ and \mathcal{L}_{del}^I since the basic actions in $\mathcal{L}_t^{I,M,-*}$ 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 van Benthem et al. (2009): If we see $\langle \mathbb{A}, e \rangle$ as a basic action modality when considering PDL semantics, then for any formula $\phi \in \mathcal{L}_{del}^I$:

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

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 ⁶. We now show that the effects of actions in $\mathcal{L}_t^{I,M}$ cannot be simulated by action models.

⁶Due to the limit of space, readers are referred to van Benthem et al. (2009) for details.

Proposition 4. *There is no translation of action models $T : A \rightarrow \mathcal{A}$ such that for all consistent $\mathcal{L}_i^{I,M}$ -states s :*

$$T(\text{Exp}T^x), \text{hist}(s) \Leftrightarrow \text{Forest}(\text{Exp}K^x(s), \mathcal{A}), (s)$$

where $x \in \{\text{set}, \text{1st}, \text{asyn}\}$, $T(\text{Exp}T^x)$ is the IS obtained from $\text{Exp}T^x$ by replacing each label of $\alpha \in A$ by $T(\alpha) \in \mathcal{A}$ and \Leftrightarrow is the bisimulation for transitions labeled by $I \cup \mathcal{A}$.

Proof. van Benthem et al. (2009) shows that $\text{Forest}(\text{Exp}K^x(s), \mathcal{A})$ 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, $\text{Exp}T^x$ clearly does not satisfy this property for $x \in \{\text{set}, \text{1st}, \text{asyn}\}$ in general. For example, $\text{send}_j^i(M); \gamma \sim_j^x \gamma; \text{send}_j^i(M)$ where $x \in \{\text{set}, \text{1st}, \text{asyn}\}$ and γ is some action j cannot observe, but $\text{send}_j^i(M) \not\sim_j^x \gamma$. \square

If we consider τ -semantics, then a correspondence result can be obtained. Let $T_{\text{DEL}} : \mathcal{L}_i^{I,M,*} \rightarrow \mathcal{L}_{\text{del}}^I$ be defined as follows:

$$\begin{aligned} T_{\text{DEL}}(\top) &= \top \\ T_{\text{DEL}}(p) &= p \\ T_{\text{DEL}}(\neg\phi) &= \neg T_{\text{DEL}}(\phi) \\ T_{\text{DEL}}(\phi_1 \wedge \phi_2) &= T_{\text{DEL}}(\phi_1) \wedge T_{\text{DEL}}(\phi_2) \\ T_{\text{DEL}}([\alpha]\phi) &= [\text{Exp}A_i^\tau(\alpha)]T_{\text{DEL}}(\phi) \\ T_{\text{DEL}}([\pi_1 \cup \pi_2]\phi) &= T_{\text{DEL}}([\pi_1]\phi) \wedge T_{\text{DEL}}([\pi_2]\phi) \\ T_{\text{DEL}}([\pi_1; \pi_2]\phi) &= T_{\text{DEL}}([\pi_1][\pi_2]\phi) \end{aligned}$$

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

- $E = \{e_\beta \mid \beta \in A\}$
- $e_\beta R_i e_{\beta'} \Leftrightarrow \iota(\beta) = \iota(\beta')$ or $i \notin \text{Obs}(\beta) \cup \text{Obs}(\beta')$.
- $\text{Pre}(e_\beta) = T_{\text{DEL}}(\text{Pre}(\beta))$.
- If $\text{Pos}(\beta) = \langle M_0, \dots, M_I, x \rangle$ then:

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

$$\text{Pos}(e_\beta)(\text{com}(G)) = \text{com}(G)$$

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

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

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

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

$$s \models^\tau \phi \Leftrightarrow \text{Exp}K^\tau(s), s \Vdash_{\text{DEL}} T_{\text{DEL}}(\phi).$$

However, without the internal structure of basic propositions and protocol constraints in action models, the translation to standard *DEL* relies on infinitely many atomic propositions and action models which change infinitely many atomic propositions.

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. Recall that we say an action α *respects the communication channel* if $\text{Pre}(\alpha) \models \text{com}(\text{Obs}(\alpha))$.

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

$$s \models^{\text{asyn}} \langle \text{exprot}(\pi) \rangle (\neg C_I \varphi \rightarrow \neg \langle \bar{\alpha} \rangle C_I \varphi)$$

Proof. Let $s \Vdash \langle \text{exprot}(\pi) \rangle t$ and suppose $t \models^{\text{asyn}} \neg C_I \varphi$. Towards a contradiction, let $\bar{\alpha}$ be the minimal sequence of actions such that $t \models^{\text{asyn}} \langle \bar{\alpha} \rangle \phi$. Let $\bar{\alpha} = \bar{\beta}; \alpha$, $t \Vdash \langle \bar{\beta} \rangle u$ and $u \Vdash \langle \alpha \rangle v$. Since $I \notin \text{CC}(s)$ and α respects the communication channel, $\text{obs}(\alpha) \neq I$ so there exists $j \notin \text{Obs}(\alpha)$. Then $AM(u)|_j^{\text{asyn}} = AM(v)|_j^{\text{asyn}}$ so $u \sim_j^{\text{asyn}} v$. Since $\bar{\alpha}$ was minimal, $u \not\models^{\text{asyn}} C_I \varphi$. But then $u \models^{\text{asyn}} \neg K_j C_I \varphi$ so $v \models^{\text{asyn}} \neg K_j C_I \varphi$. So $v \not\models^{\text{asyn}} C_I \varphi$. \square

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 τ -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 τ -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 α , if $s \sim_i^\tau t$ and we have s', t' such that $s \llbracket \alpha \rrbracket s'$ and $t \llbracket \alpha \rrbracket t'$ 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 $s' \sim_i^\tau t'$. \square

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 $s \llbracket \alpha \rrbracket s'$ and $t \llbracket \alpha \rrbracket t'$. This means that s and t both satisfy the preconditions of α , so essentially no knowledge that distinguishes s and t is introduced by α .

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

$$\phi ::= has_{im} \mid com(G) \mid \neg\phi \mid \phi_1 \wedge \phi_2$$

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 α be an action that does not change the agent’s message sets or the protocol. For any $\phi \in \mathcal{L}_{bool}$ and any state s : $s \models \phi \leftrightarrow \langle \alpha \rangle \phi$.*

Combining the properties of the actions from the previous lemma, we call an action α_d^G to be a *dummy action* for a group of agents G if its internal structure has the precondition $com(G) \wedge future(\alpha_d^G)$, 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 α_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 $\bar{\alpha}$,*

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

Proof. Suppose towards a contradiction that $s \models \neg C_I \phi$ and there is a minimal sequence $\bar{\alpha}$ such that $s \models^\tau \langle \bar{\alpha} \rangle C_I \phi$. Let $\bar{\alpha} = \bar{\beta}; \alpha$ and let $i \notin Obs(\alpha)$. Such i always exists since $I \notin CC(s)$. Let α_d^G be a dummy action such that $i \notin G$. Let $s \models \bar{\beta} u$. Since $\bar{\alpha}$ is minimal, $u \models^\tau \neg C_I \phi$, so there is a \sim_I -path from u to a world t such that $t \not\models^\tau \phi$. Since it is common knowledge that any action in A is possible, then we can execute α_d^G at any world on the path from u to t . By lemma 1 α_d^G preserves the relations between states so there are states u', t' such that $u \models \alpha_d^G u'$, $t \models \alpha_d^G t'$ and $u' \sim_I t'$. Also, since $t \not\models^\tau \phi$ and by lemma 2, $t' \not\models^\tau \phi$. So $u' \not\models^\tau C_I \phi$. This means that if we would execute α_d^G in state u , then $C_I \phi$ would not hold.

Let $u \models \alpha_d^G u'$ and $u \models \alpha v$. Because $i \notin G$, i cannot see the difference between executing α_d^G and α : $AM(u')|_i^\tau = (AM(u)|_i^\tau; \tau) = AM(v)|_i^\tau$ so $u' \sim_i^\tau v$. But we just saw that $u' \not\models^\tau C_I \phi$, so then $v \not\models^\tau C_I \phi$. But this contradicts our assumption that $\bar{\beta}; \alpha$ induced common knowledge of ϕ . \square

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 ϕ and protocols π in $\mathcal{L}_i^{I,M}$.
4. Use *exinfo*(ϕ) and *exprot*(π) 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 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:

$$\begin{aligned} call_j^i(M') &:= \bigcup_{M'' \subseteq M'} shareall_{\{i,j\}}(M'') \\ mail_j^i(M') &:= \bigcup_{M'' \subseteq M'} sendall_{\{ij\}}^i(M'') \end{aligned}$$

Here $call_j^i(M')$ is the call between agent i and j where they share all messages out of M' they possess⁷. 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 $mail_j^i(M')$, 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 any formula from the language instead of only their messages. This is modeled by $inform_j^i(\phi)$. Let $M_I = \{m_0, \dots, m_{|\Pi|}\}$ be the set of all secrets. For suitable finite sets of formulas Φ and protocols Π ⁸, we define

$$A = \bigcup_{\phi \in \Phi} exinfo(\phi) \cup \bigcup_{\pi \in \Pi} exprot(\pi) \cup \bigcup_{i,j \in I} call_j^i(M_I) \cup \bigcup_{i,j \in I} mail_j^i(M_I) \cup \bigcup_{i,j \in I, \phi \in \Phi} inform_j^i(\phi),$$

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 $net_I^{tel} = \{\{i, j\} \mid i \neq j \in I\}$. Then the initial state is:

$$s_I^{tel} = \langle net_I^{tel}, \{m_0\} \dots \{m_{|\Pi|}\}, \epsilon, \{m_0\} \dots \{m_{|\Pi|}\}, (\Sigma A)^* \rangle$$

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 $\pi_{call} := (\bigcup_{i,j \in I} call_j^i(M_I))^*$, $\pi_{mail} := (\bigcup_{i,j \in I} mail_j^i(M_I))^*$ as the protocols where the agents can only make telephone calls or send voicemails, respectively. We define $\pi_{call, inform} := (\bigcup_{i,j \in I} call_j^i(M_I) \cup \bigcup_{i,j \in I} mail_j^i(M_I))^*$.

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

⁷Here M' encodes the *relevant context* e.g. messages that are "about work".

⁸For example, the sets of formulas/protocols up to the length of certain large number.

network is common knowledge. We use the following abbreviations:

$$\begin{aligned} \text{OneSecEach}_I &:= \bigwedge_{i \in I} (\text{has}_i m_i \wedge \bigwedge_{j \neq i} \neg \text{has}_j m_i) \\ \text{TP} &:= \text{exinfo}(\text{com}(\text{net}_I^{\text{tel}})) \wedge \text{OneSecEach}_I \\ \text{TP}_x &:= \text{TP}; \text{exprot}(\pi_x) \\ \text{HasAll}_I &:= \bigwedge_{i \in I} \text{has}_i M_I \end{aligned}$$

OneSecEach_I states that every agent has one secret known only to him. TP_x is the action of announcing the assumptions of the scenario and protocol π_x . HasAll_I expresses that every agent knows every secret, which is the goal we want to reach.

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 ϕ 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 ϕ with the current protocol. The external actions do not abide by the protocol, so we should not consider them⁹.

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

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

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

A proof of this proposition is given in Hurkens (2000). 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: $\text{call}_5^4(M_I); \dots; \text{call}_{|I|}^4(M_I); \text{call}_2^1(M_I); \text{call}_4^3(M_I); \text{call}_3^1(M_I); \text{call}_4^2(M_I); \text{call}_5^4(M_I); \dots; \text{call}_{|I|}^4(M_I)$

⁹Note that $\langle \rangle^{\leq n}$ serves as a generalization of the *arbitrary announcement* that is added to DEL in Ågotnes et al. (2009).

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?

Intuitively we can use $mail_j^i(M_I); mail_i^j(M_I)$ to mimic each $call_j^i(M_I)$, thus we have:

$$s_I^{tel} \models^x \langle TP_{mail} \rangle \langle \rangle^{\leq 4|I|-8} HasAll_I.$$

However, we can do much better:

Proposition 6. *For any $x \in Sem$:*

$$s_I^{tel} \models^x \langle TP_{mail} \rangle \langle \rangle^{min(2|I|-2)} HasAll_I$$

Proof. Consider the following protocol: $mail_2^1(M_I); mail_3^2(M_I); \dots; mail_{|I|}^{|I|-1}(M_I); mail_1^{|I|}(M_I); mail_2^{|I|}(M_I); \dots; mail_{|I|-1}^{|I|}(M_I)$. 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 $2|I| - 2$. \square

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 $HasAll_I$. If there are only three agents, this is possible by making telephone calls:

Proposition 7. *If $|I| \leq 3$ then for some $n \in \mathbb{N}$:*

$$s_I \models^{\tau} \langle TP_{call} \rangle \langle \rangle^{\leq n} C_I HasAll_I$$

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 $call_2^1(M_I); call_3^2(M_I); call_1^2(M_I)$. After execution of this protocol all agents know all secrets, and agent 2 knows this. Also, since agent 1 learned the secret of agent 3 from agent 2, he knows that

agent 2 and 3 must have communicated after the last time he spoke to agent 2, so agent 3 must know the secret of agent 1. Regarding agent 3, he knows agent 2 has all secrets the moment he communicated with agent 2, and he observed a τ when agent 2 called agent 1 after that. Since there are only three agents agent 3 can deduce that agent 1 and 2 communicated so he knows agent 1 knows all secrets. Since all agents can reason about each others knowledge it is common knowledge that all agents have all secrets. \square

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. In the above results the communicative power of the agents is still fairly limited. They can only communicate their messages and they cannot talk about higher-order knowledge. An 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. Interestingly, this reduces the possibilities to reach common knowledge since the dummy action $inform_C^i(\tau)$ is allowed. The agents have no clue whether any information is transferred when they observe a τ action so they can never reach common knowledge, not even in the case that $|I| = 3$. This directly follows from Theorem 4.

Proposition 8. *For any $n \in \mathbb{N}$, if $|I| > 2$ then:*

$$s_I \not\models^\tau \langle TP_{call,inform} \rangle \langle \rangle^{\leq n} C_I HasAll_I$$

However, we can approach common knowledge arbitrarily close. For any finite sequence of agents $w = ij\dots k$ define:

$$K_w \varphi := K_i K_j \dots K_k \varphi$$

Proposition 9. *For any finite sequence w of agents from I , there exists some $n \in \mathbb{N}$ such that:*

$$s_I \models^\tau \langle TP_{call,inform} \rangle \langle \rangle^{\leq n} K_w HasAll_I$$

Proof. We will give a protocol that results in the desired property. First we execute the protocol given in the proof of Proposition 6. Note that after executing this protocol, agent $|I|$ knows that everyone knows all secrets. Let $w = a_1 \dots a_n$. We execute $inform_{a_n}^{|I|}(HasAll_I)$; $inform_{a_{n-1}}^{|I|}(K_{a_n} HasAll_I)$; \dots ; $inform_{a_1}^{|I|}(K_{a_2} \dots K_{a_n} HasAll_I)$ and clearly, after these actions the desired property will hold. \square

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 τ -semantics, it is possible to reach common knowledge:

Proposition 10. *There is a protocol π of call actions such that*

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

Proof. Let π be the protocol given in the proof of proposition 5. Since each agent observes a τ 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. \square

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 Dehaye et al. (2003) or many situations in distributed computing.

However, when we use *asyn*-semantics, the agents cannot count the number of communicative actions happening, so they never know when the protocol has been executed. Because of this they can never reach common knowledge:

Proposition 11. *There is no protocol π of call and inform actions such that*

$$s_I \models^{\text{asyn}} \langle TP; \text{exprot}(\pi) \rangle \langle \rangle^{\leq n} C_I \text{HasAll}_I$$

Proof. Follows from Theorem 3. \square

These results show the way we can use our framework to model a lot of different situations, often with surprising outcomes.

5 Conclusions and Future work

We developed an expressive dynamic epistemic logic tailored to specify and reason 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 Pacuit and Parikh (2007) : “ i gets j ’s information without j noticing that” as $\alpha = \text{download}_j^i(M)$ with $\text{Obs}(\alpha) = i$, $\text{Pre}(\alpha) = \text{com}(\{i, j\}) \wedge \text{has}_j M$ and a suitable postcondition adding messages to i ’s information set¹⁰. 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
Roelofsen (2005)	inform	propositions	\equiv^r
Pacuit and Parikh (2007)	download	Boolean atomic propositions	\equiv^r
Apt et al. (2009)	inform	positive atomic propositions	\equiv^{set}
Our work	by design	messages or formulas	by design

We end with a list of further issues to be explored:

Theoretical Issues Many theoretical issues are left for future work e.g. the model checking and satisfiability problem of (the fragments of) $\mathcal{L}_i^{I,M}$ w.r.t different x -semantics and the expressivity of $\mathcal{L}_i^{I,M}$ compared to various fixed point logics. Another interesting issue is the logical characterization of the observational equivalences defined in our work.

Network In this work, we take the hypergraphs of Apt et al. (2009) as networks, thus assuming the communication channels to be symmetric. More constrained network definitions with asymmetric channels are also possible. Moreover, different social networks/organizations may have different properties, e.g. the network of a group of gossiping girls is usually connected and transitive¹¹ while the network of a secret society is usually not transitive due to a hierarchy and secrecy. Thus leaking a secret to your closest girl friend may cause it to be a shared knowledge among all the girls on the next day, but gossiping about your boss with the juniors under your supervision might be safe in a secret society.

Actions There are other useful actions that we did not cover in this paper. For example, we have assumed that message passing actions are always monotonic, but there are cases when deleting messages from memory or buffer is natural. Another assumption is that the agents either clearly observe an action or observe nothing at all. This excludes the modeling of actions which may give some agents partial observations e.g. BCC in email. Roelofsen (2005) also

¹⁰Pacuit and Parikh (2007) phrases such download action with propositions instead of messages.

¹¹In the sense that if girl A can call girl B and girl B can call girl C then A is in touch with C.

mentioned the possibility of changing the channels, e.g. deleting people from your Christmas card sending list if they did not reply to your card last year. Furthermore, the actions that change the protocol may also need to be constrained by the communication channels, as discussed in Moses et al. (1986). This will raise more interesting issues, e.g., whether different levels of knowledge of the protocol (weaker than common knowledge) suffice to facilitate the successful runs of certain class of protocols. Such actions could be handled within our framework with little adaption.

Protocol We use regular expressions without tests to specify sequential protocols. We leave out tests since the observation of a test is not clear, unless it is grouped with follow-up actions. It seems that this is expressive enough for many useful applications. In a more general setting, we would like to have tests and parallel composition in the protocol language and model the protocol by composing local protocols for each agent.

Knowledge Transfer Our framework paves a way to discuss message passing and knowledge transfer over communication channels at the same time. It may be applicable to a security setting where information flow should be controlled strictly complying certain knowledge requirements.

Acknowledgements We would like to thank Krzysztof Apt and Johan van Benthem for the detailed discussions and comments on earlier versions of this work, and also Alexandra Silva for pointing out the references to the derivatives of regular expressions. We also thank the anonymous referees of AAMAS10 for their insightful comments.

References

- K. R. Apt, A. Witzel, and J. A. Zvesper. Common knowledge in interaction structures. In A. Heifetz, editor, *TARK*, pages 4–13, 2009.
- A. Baltag and L. Moss. Logics for epistemic programs. *Synthese*, 139(2):165–224, March 2004.
- A. Baskar, R. Ramanujam, and S. P. Suresh. Knowledge-based modelling of voting protocols. In *TARK '07: Proceedings of the 11th conference on Theoretical aspects of rationality and knowledge*, pages 62–71, New York, NY, USA, 2007. ACM.
-

- J. A. Brzozowski. Derivatives of regular expressions. *J. ACM*, 11(4):481–494, October 1964. ISSN 0004-5411.
- M. Cohen and M. Dam. A complete axiomatization of knowledge and cryptography. In *LICS*, pages 77–88. IEEE Computer Society, 2007.
- J. H. Conway. *Regular Algebra and Finite Machines (Chapman and Hall mathematics series)*. Chapman and Hall, September 1971. ISBN 0412106205.
- P. O. Dehaye, D. Ford, and H. Segerman. One hundred prisoners and a light bulb. *Mathematical Intelligencer*, 24(4):53–61, 2003.
- R. Fagin, J. Y. Halpern, M. Y. Vardi, and Y. Moses. *Reasoning about knowledge*. MIT Press, Cambridge, MA, USA, 1995.
- J. Gerbrandy and W. Groeneveld. Reasoning about information change. *Journal of Logic, Language and Information*, 6(2):147–169, April 1997.
- T. Hoshi. *Epistemic Dynamics and Protocol Information*. PhD thesis, 2009.
- T. Hoshi and A. Yap. Dynamic epistemic logic with branching temporal structures. *Synthese*, 169(2):259–281, July 2009.
- C. A. J. Hurkens. Spreading gossip efficiently. *Nieuw Archief voor Wiskunde*, 5/1(2):208–210, 2000.
- Y. Moses, D. Dolev, and J. Y. Halpern. Cheating husbands and other stories: A case study of knowledge, action, and communication. *Distributed Computing*, 1(3):167–176, September 1986. ISSN 0178-2770.
- E. Pacuit and R. Parikh. Reasoning about communication graphs. In J. van Benthem, D. Gabbay, and B. Löwe, editors, *Interactive Logic — Proceedings of the 7th Augustus de Morgan Workshop*, Texts in Logic and Games, pages 135–157, Amsterdam, 2007.
- R. Parikh and R. Ramanujam. A knowledge based semantics of messages. *Journal of Logic, Language and Information*, 12(4), 2003.
- R. Parikh and R. 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. ISBN 3-540-15648-8.
- T. Ågotnes, P. Balbiani, H. van Ditmarsch, and P. Seban. Group announcement logic. *Journal of Applied Logic*, July 2009. ISSN 15708683.
-

-
- R. Ramanujam and S. P. Suresh. Deciding knowledge properties of security protocols. In *Proc. Theoretical Aspects of Rationality and Knowledge*, pages 219–235. Morgan Kaufmann, 2005.
- F. Roelofsen. Exploring logical perspectives on distributed information and its dynamics. Master’s thesis, University of Amsterdam, 2005.
- N. V. Shilov and N. O. Garanina. Model checking knowledge and fixpoints. In Z. Ésik, A. Ingólfssdóttir, Z. Ésik, and A. Ingólfssdóttir, editors, *FICS*, volume NS-02-2 of *BRICS Notes Series*, pages 25–39. University of Aarhus, 2002.
- J. van Benthem. ‘one is a lonely number’: on the logic of communication. In Z. Chatzidakis, P. Koepke, and W. Pohlers, editors, *Logic Colloquium ’02*, pages 96–129, Wellesley MA, 2002. ASL & A.K. Peters.
- J. van Benthem, J. van Eijck, and B. Kooi. Logics of communication and change. *Information and Computation*, 204(11):1620–1662, November 2006.
- J. van Benthem, J. Gerbrandy, T. Hoshi, and E. Pacuit. Merging frameworks for interaction. *Journal of Philosophical Logic*, 38(5):491–526, October 2009. ISSN 0022-3611.
- R. van der Meyden and N. Shilov. Model checking knowledge and time in systems with perfect recall. In *Foundations of Software Technology and Theoretical Computer Science*, pages 432–445, 1999.
- H. van Ditmarsch. *Knowledge games*. PhD thesis, 2000.
- H. van Ditmarsch, W. van der Hoek, and B. Kooi. *Dynamic Epistemic Logic*. (Synthese Library). Springer, 1st edition, November 2007. ISBN 1402069081.
- Y. Wang, L. Kuppusamy, and J. van Eijck. Verifying epistemic protocols under common knowledge. In *TARK ’09: Proceedings of the 12th Conference on Theoretical Aspects of Rationality and Knowledge*, pages 257–266, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-560-4.
-