The logic of Gossiping

Hans van Ditmarsch, Wiebe van Der Hoek, Louwe Kuijer

The Logic of Gossiping

Hans van Ditmarsch Wiebe van der Hoek Louwe B. Kuijer

Abstract

The so-called gossip problem is a formal model of peer-to-peer communication. In order to perform such communication efficiently, it is important to keep track of what agents know about who holds what information at a given point in time. The knowledge that the agents possess depends strongly on the particular type of communication that is used.

Here, we formally define a large number of different variants of the gossip problem, that differ in the extent to which communication is private (observable, synchronous or asynchronous), the direction of the flow of information (caller to callee, callee to caller or both) and whether the agents become aware of the exact set of information possessed by their communication partner.

We consider a number of formulas that represent interesting properties that a gossip situation may or may not enjoy, and show for which variants they are valid. Additionally, we show that the model checking and validity checking problems for each variant are decidable, and we introduce sound and complete proof systems for them.

1 Introduction

1.1 The Gossip Problem

The *gossip problem* [36, 10, 25] models the spread of information through peer-to-peer communication. In the gossip problem, there is a set Ag of agents of size $|Ag| = n$, each of which starts with a private piece of information that is referred to as that agent's *secret*. An agent who knows the secrets of every agent is called an *expert*. The agents' goal is to share information in such a way that they all become experts, and to do so as efficiently as possible. In order to achieve this goal, the agents communicate using one-to-one interactions, usually referred to as *telephone calls* or simply *calls*. Importantly, when agents communicate they transmit not only their own secret, but also all other secrets that they have learned. So if in the first call agent a tells b its secret, then in a subsequent call between b and c, b will tell c the secrets of both a and b.

The gossip problem is usually represented by the metaphor of a number of people exchanging secrets by gossiping over the phone. It is applicable far beyond this metaphor, however, and provides a way to study any dissemination of information in distributed environments. In order to represent the many different ways to communicate, we do have to make some small modifications to the gossip problem based on the particular communication method that we wish to model. For example, in some cases

the caller tells all its secrets to the callee without hearing anything in return, while in other cases both caller and callee tell their secrets. This way a large number of different variants of the gossip problem can be defined, one for each way to communicate.

The first wave of papers about the gossip problem was published in the 1970s and 1980s by Tijdeman, Baker, Hajnal and many others [36, 10, 23, 13, 35]. These papers typically focused on establishing, for a given variant, the minimum number of calls needed to turn all agents into experts. In the most commonly studied variant, where both agents in a call tell each other all the secrets they know, the minimum number of calls is $2n - 4$, assuming that $n \geq 4$ [36, 10, 23].

Unfortunately, the solutions where all agents become experts after $2n - 4$ calls are very hard, and sometimes even impossible, to achieve in a distributed setting. This is because such solutions require a large amount of coordination. If there is a central authority that instructs each agent who to call and when to do so, then $2n - 4$ can be achieved. But in a distributed system such a central authority is typically absent (one could even argue that, by definition, a distributed system lacks such a central authority). Without anyone telling them exactly what to do, the agents are left to do the best they can based on their (possibly rather limited) knowledge of the situation. Often, the agents cannot coordinate sufficiently to make everyone experts in $2n - 4$ calls.

The failure to reach a sufficiently high level of coordination for the $2n - 4$ solution has lead to many papers on the gossip problem where calls are made randomly [19, 22, 27, 25, 28], and more recently to a new wave of papers on the gossip problem that study *epistemic protocols* [6, 1, 37, 7, 26, 3, 40, 4, 17]. A protocol, in this context, is simply a list of instructions of the form "if the condition φ_{ab} is true, agent a should call agent b". Because agents need to base their actions on their knowledge, not all such instructions can be carried out, however. If a doesn't know whether the condition φ_{ab} holds, it cannot determine whether it should call b. A protocol is *epistemic* if the agents can obey their instructions, so if whenever φ_{ab} is true, a knows that φ_{ab} is true.

An epistemic protocol can only depend on things that the agents know, so in order to design such a protocol we first have to determine which things the agents know. This task is complicated by the fact that what agents know depends on the type of communication that is used, so the agents' knowledge depends on the specific variant of the gossip problem that we are considering. For example, take the following protocol, in the commendable survey by Hedetniemi *et al.* [25] mentioned as the protocol NOHO, for 'No One Hears Own':

> **Tell new secrets (TNS)**
> Agent a may call agent b if b doesn't know a's secret.

So the condition φ_{ab} is given by "b doesn't know a's secret." Note that there may be multiple pairs (a, b) for which the condition φ_{ab} holds. In that case, there are multiple calls that are possible. Which of these possible calls actually happens is considered to be non-deterministic.

In most gossip variants, TNS is not an epistemic protocol, since a typically cannot be certain that b doesn't know a's secret. After all, once a tells its secret to anyone else, a loses control of where it spreads; if a tells its secret to c, then c might tell it to d who might tell it to b. But suppose now that the agents are using a communication type where the meta-data of each call is public, i.e., where every agent can see exactly

who has called whom. In that variant, a can keep track of where its secret has spread: if a tells its secret to c who tells it to b, then a will see that the cb call has taken place. So, in particular, whenever b doesn't know a's secret, a knows this. The protocol TNS is therefore epistemic in this variant, even if it is non-epistemic in most other variants.

There are a number of protocols that are epistemic in every variant of the gossip problem. Consider, for example, the variant of TNS known as *hear my secret* [1]:

> **Hear my secret (HMS)**
> Agent a may call agent b unless a knows that b knows a's secret,

and the protocol *learn new secrets* [6]:

> **Learn new secrets (LNS)**
> Agent a may call agent b if a doesn't know b's secret.

The protocols HMS and LNS are epistemic regardless of the type of communication. Furthermore, they are *strongly successful*, i.e., they are guaranteed to terminate and to turn all agents into experts. Unfortunately, they are not very fast: both HMS and LNS require, in the worst case, quadratically many calls, and are likely to have the same the $n \log n$ expectation as the protocol of just making random calls [32, 38]. It is unclear whether we can do substantially better than an $n \log n$ expected termination, but it seems conceivable that more efficient epistemic protocols might result by exploiting the specific properties of knowledge in gossip protocols. This requires great precision on the notion of knowledge, including its computational aspects.

> *Understanding the properties of knowledge in gossip protocols, and axiomatizing such logics, is the main subject of this paper.*

1.2 Our contribution

Let us now describe in some detail how we will model the properties of knowledge in gossip, how we will address some computational issues of decidability, and what the novelties are of our axiomatizations.

We start by formally defining 3 parameters that lead to 18 different variants of the gossip problem We then discuss a number of properties, such as perfect recall, that some variants have but others do not (Proposition 3.11). We also show that the model checking and validity checking problems for each variant are decidable (Theorem 3.20). An important consequence of this decidability is that the problems of protocol termination and success (i.e., will every agent know all secrets when the protocol terminates) are also decidable. Finally, we provide axiomatizations for all variants.

The variants that we discuss differ in three parameters.

Firstly, there is the level of *privacy* that a call enjoys. The lowest level of privacy is the one that we described above, where the meta-data of each call is *public*. An intermediate level of privacy is if the communication is *synchronous*, i.e., if all agents notice it when a call happened but they don't know the identity of the two agents involved. This is the most commonly investigated level of privacy in the distributed computing community of gossip protocols, where calls are made in rounds. A *round*

is such a moment of synchronization, sometimes of multiple calls, but in our setting of sequential gossip every round consists of a single call. At the highest level of privacy calls are *asynchronous*, so the agents are completely unaware of any calls they are not involved in.

Secondly, we distinguish different variants with respect to the direction of communication. Calls can be of the type *push* where the caller tells its secrets to the callee, of the type *pull* where the callee tells its secrets to the caller or of the type *push-pull* where both agents tell each other their secrets.

Finally, we distinguish communication where agents tell their partner all secrets, from communication where agents tell only those secrets that their partner does not yet know. We refer to this final parameter as the level of *observance*.

In Section 2 we discuss the three parameters in more depth, and define the language that we use to reason about the knowledge of agents is gossip situations. Following that, in Section 3, we define the models that describe the various types of gossip problem, and use them to define the semantics for the language. In Section 3 we also discuss the first few results about the differences between the variants, and prove the decidability of model checking and validity checking. In the main Sections 4–6 we introduce axiomatizations for knowledge in all 18 variants, using a modular proof system that contains so-called *reduction axioms*. Before all of that, however, we first mention some background and related work, which we will also recall in greater technical detail in the final concluding Section 7.

1.3 Related Work

In this subsection we investigate the gossip literature, some of which was already mentioned in the previous subsection, in greater detail.

The gossip problem constitutes an excellent model to study information dissemination in distributed environments. The literature in distributed protocols has therefore taken up the problem and analyzed it together with a wealth of variations including different communication primitives (e.g., broadcasting instead of one-to-one calls), as well as communication structures (networks), faulty communication channels [14] and probabilistic information transmission, where the spreading of gossips is used to model the spread of epidemics [9, 34]. Surveys are [22, 27, 25, 28].

Epistemic protocols can be carried out by agents in a distributed system, so they are distributed protocols. More specifically, their reliance on the knowledge of agents makes epistemic protocols examples of so-called knowledge-based protocols as studied in distributed systems [33, 30, 24, 21]. Such distributed systems have also been object of study from the perspective of epistemic logic [20, 31], which has provided a useful level of abstraction from which to address a number of problems related to distributed computing and communication [33, 30, 21], such as protocols for the sequence transmission problem [24].

Of the publications on epistemic protocols, Attamah *et al.* [6] modeled a synchronous setting wherein all agents know whether a call took place, although if they are not involved in that call they may not know between whom the call took place. In that paper its authors also modeled the semantics of calls in *dynamic epistemic logic* [39], with reduction properties for knowledge after calls that are similar to the ones we propose in this work. Apt *et al.* [1] study the purely distributed setting, where agents

are unaware of any calls that they are not directly involved in. Additionally, [1] modeled three different directions of communication: the caller and callee tell each other their secrets (push-pull), the caller tells the callee their secrets but does not hear any secrets in return (push) or the callee tells the caller their secrets (pull). The paper also discusses gossip protocols for specific networks, like the 'Ring Protocol', for the circular gossip graph wherein agents can only call their left and right neighbors. Gossip can also be seen as an instance of multi-agent epistemic planning. In [16], this was studied using the planning language PDDL.

Herzig and Maffre [26] studied gossip protocols where the aim was not merely to turn everyone into experts (level 1 shared knowledge), but also to achieve that everybody knows that everyone is an expert (level 2 shared knowledge), and so on: higher-order knowledge of the fact that everyone is an expert (level k shared knowledge). They presented a protocol that achieves k-level shared knowledge in $(k+1)(n-2)$ steps. It should be noted that this requires that not only secrets are exchanged in messages but also knowledge, including higher-order knowledge, of secrets. Without that, level 2 shared knowledge can still be achieved but it seems unlikely that level k shared knowledge, for $k > 2$, can then be achieved.

Van Ditmarsch *et al.* [41, 40] present studies of *dynamic* gossip protocols, where in calls the agents not only share their secrets but also the set of their *neighbours*, i.e., their network links to other agents. The purpose of these works is to characterize such protocols in terms of the class of graphs for which they terminate. In turn, in [37] the gossip problems are presented in an epistemic framework that provides several parameters allowing one to capture such aspects as the initial knowledge of the agents, the type of communication used (such as the above direction of the information exchange: pushpull, push, and pull), and the desired type of the protocol. For some of the combinations of the parameters the minimum number of calls needed to reach the final situation is then established. In [38] it is investigated which distributions of secrets can be reached by particular epistemic gossip protocols, and reports on simulations, calculations, and (information theoretical) approximations of expected protocol execution time, including achieving the $O(n \log n)$ expected complexity of termination for various other protocols than the one where all calls are random.

The papers most directly related to this one are [37], [2] and a series of papers by Apt and Wojtczak [3, 4, 5]. In [37], some of the key information assumptions on calls were identified—most notably assumptions about synchrony/asynchrony. A main other topic in that publication was the minimum number of calls under different conditions of coordination. In [2] the difference between synchronous and asynchronous communication was also discussed, along with two more parameters: direction (push, pull or push-pull) and observance (before or after).[1] We consider the same set of parameters as in [2], and several of the definitions we use in this paper are also identical to the ones in [2]. The papers [3], [4] and [5] also build upon the definitions of [2] but use a more restricted set of parameters. Specifically, they vary only the direction of the call (push, pull or pushpull) while leaving the levels of privacy and observance constant.[2] Furthermore, they only consider the depth 1 fragment of epistemic logic.

[1] See Section 2 for definitions of synchrony, call direction and observance.

[2] Using the notation introduced in Section 2, [3, 4, 5] consider only the call types $(\bullet, \mathsf{d}, \alpha)$.

This means that a call condition φ_{ab} cannot depend on higher order knowledge. So, for example, a call condition where a can call b if a knows that b doesn't know whether c knows a's secret is not allowed in that setting. For this restricted set of parameters and this fragment of the language, [3] and [4] show that the model and satisfiability checking problems are decidable. They also establish several computational complexity results for those problems and a number of related problems. In [5] a number of open questions regarding epistemic gossip are introduced.

The decidability results proven in this paper extend the ones from [3, 4] by considering the full set of parameters introduced in [2], as opposed to varying only the direction of calls, and considering the full language, as opposed to only the depth 1 fragment. Interestingly, considering the full language also makes the difference between the three directions much more pronounced; the proofs in [3, 4] for the three directions are completely analogous, while our results for push-pull differ significantly from our results for push and pull. (See Section 5 for details.) We also consider arbitrary starting situations for gossip, while [3, 4, 5] only consider the unique starting situation where it is common knowledge that every agent initially only knew its own secret. Finally, we introduce axiomatizations for each variant of the gossip problem, thereby answering the first two open questions from [5]. Along the way we also show that common knowledge does not reduce to nested knowledge, see Remark 5.9, which answers the third open question from [5].

2 Preliminaries and Syntax

Throughout the paper we assume a fixed finite set Ag of at least three *agents*. We assume that each agent has exactly one *secret* and that the secrets are pairwise different. Each secret is viewed as a distinct symbol. We denote by S the set of all secrets and the secret of agent a by A, the secret of agent b by B and so on. If agent a has found out the secret B of b, we say that a *is familiar with* B, and write F_aB.

Furthermore, we assume that each secret carries information identifying the agent to whom this secret belongs. So once agent a learns secret B she knows that she learned the secret of agent b. An agent is familiar with their own secret and all the secrets they learned by making calls.

2.1 Calls

Calls constitute the sole form of knowledge acquisition the agents have to their disposal. In its abstract form, a call c is a pair of agents ab (with $a \neq b$). Each call concerns four roles, *caller*, *callee*, *informer* and *listener* (we will come back to these roles shortly). The agents a and b with $a \neq b$ that constitute call c are denoted by $Ag(\mathsf{c}) = \{a, b\}$. Any agent c different from a and b is called an *outsider* of the call. Our account of calls takes into account various aspects of them, namely:

- *privacy*, which is concerned with what outsiders note about the call,
- *direction*, which clarifies the direction of the information flow in the call,

- *observance*, which clarifies, when an agent a is informed by b, whether a sees b's secrets before fusing them with her own, or only sees the result of the fusion of the two sets of secrets.

Each of the choices for the above parameters constitutes a call type τ. More formally, a call type $\tau = (\mathsf{p}, \mathsf{d}, \mathsf{o})$ where, in each type, the parameter p ranges over $\mathsf{P} = \{\circ, \text{◓}, \bullet\}$ and is called the *privacy level*, or simply *privacy*, $\mathsf{d} \in \mathsf{D} = \{\diamond, \triangleleft, \triangleright\}$, is called the *direction type* (or *direction*) of the call, and $\mathsf{o} \in \mathsf{O} = \{\alpha, \beta\}$, is called the *observance level* (or *observance*). Often, the call type (or parts of it) is (are) clear from the context, and we omit it (them). In our examples, at the level of calls, we often only explicitly mention the direction type, and they may be written infix, i.e., $a \triangleright b$ means $ab^{\triangleright}$, etc. For a type τ like $(\circ, \diamond, \beta)$, we define $\tau(\mathsf{p}) = \circ, \tau(\mathsf{d}) = \diamond$, etc. Note we have now defined 18 different types of calls.

In theory, it would be possible for different call types to be combined; a call of type $(\circ, \diamond, \alpha)$ could be followed by a call of type $(\text{◓}, \triangleright, \beta)$. We are not aware of anyone studying such mixed call types, however, nor do we consider mixing call types to be particularly useful. We therefore consider the call type to be a global parameter, i.e., in any given situation we consider the call type to be fixed.

If $Ag(\mathsf{call}) = \{a, b\}$, the privacy type determines the extent to which an agent $c \notin \{a, b\}$ notices call taking place.

- $\circ$: every agent c, even those $c \neq a, b$, notes that c happens,
- ◓: every agent $c \neq a, b$ notes that *some* call takes place, though not between whom,
- $\bullet$: no agent $c \neq a, b$ notes that a call is taking place.

We refer to any call type τ with $\tau(\mathsf{p}) \in \{\circ, \text{◓}\}$ as a synchronous call, while $\tau(\mathsf{p}) = \bullet$ denotes the asynchronous case. Intuitively, those degrees can be ordered as $\circ <_{\mathsf{p}}$ ◓ $<_{\mathsf{p}} \bullet$, with $\circ$ meaning no privacy at all, and $\bullet$ denoting full privacy. Conversely, from the perspective of the agents not involved in the call, a call with privacy level $\circ$ is the most informative, while calls of privacy level $\bullet$ are the most opaque.

Next, we distinguish three *direction types*, in short *directions*, of a call:

- *push-pull*, written as $\diamond$.

  As a result of the call $a \diamond b$ the caller a and the callee b learn each other's secrets (and they each play the role of *informer* and *listener*).

- *push*, written as $\triangleright$.

  As a result of the call $a \triangleright b$, the callee b learns all the secrets held by the caller a. Agent b is the listener, a the informer.

- *pull*, written as $\triangleleft$.

  As a result of the call $a \triangleleft b$, the caller a learns all the secrets held by the callee b. Agent b is the informer, a the listener.

The directions $\triangleright$ and $\triangleleft$ have many properties that are 'symmetric', but there is one important distinction. Regardless of the direction type, it is always the caller who initiates a call. So in a call $a \triangleright b$ agent a decides to give information to b, while in a call $a \triangleright b$ agent a decides to take information from b. In this paper, that difference does not come up much, since we are interested in modeling the information effects of calls, and the information gained by a and b in $a \triangleright b$ is exactly the same as that gained in $b \triangleleft a$. But remember that our goal is to use the agents' knowledge to define an epistemic gossip protocol. In such a protocol, every call has a call condition, and a call can be placed if the caller knows that this condition holds. So for the call $a \triangleright b$ to happen agent a must know that the condition holds, while for the call $b \triangleleft a$ it is b that must know it.

Depending on the direction of a call between a and b there are one or two agents who can learn new *secret*, although both agents may learn new *information*: for $a \triangleright b$ for instance, b learns all of a's secrets, and a learns that b has learned them. Given a direction d, we denote the set of calls with that direction as C^{d}.

We finally consider two possible levels of *observance* of a call:

- *after*, written as α: During the call the listener(s) incorporate the secrets of their informer with their own secrets, and only after that, inspect the result.
- *before*, written as β: During the call the listener(s) inspect the secrets of their informer before adding them to their own secrets.

The difference between the two levels of observance is quite subtle, but it can be intuitively understood as follows. Think of the secrets known to an agent as a folder containing a file for each known secret. If the observance level is β, the informer either sends a copy of their secrets folder to the listener, or gives the listener read-only access to the folder. The listener can then inspect the folder, thereby discovering exactly which secrets are known to the informer, and makes copies of those secrets they didn't already know.

If the observance level is α, the informer is given write-only access to the listener's folder of secrets. The informer simply merges their folder with that of the listener, without checking the result of the merger. If a secret A was unknown to the listener before but known after this merger, the listener can conclude that the informer knows A. If B remains unknown to the listener after the merger then the listener can conclude that the informer doesn't know B, or at least didn't know it at the time of the call. But if C was already known to the listener before the merger, they gain no information about whether the informer knows C.

Example 2.1.

a First, assume the call type is $(\bigcirc, \diamond, \alpha)$. Consider the initial situation, i.e., where everybody is only aware of their own secret, and assume that all agents know we are in the initial situation. Assume moreover that first a call ab takes place, and then a call bc. Let us now reason from the perspective of agent d. Because the privacy level is $\bigcirc$, after the first call, agent d knows that both a and b are now familiar with A and B. This then implies that d also knows that after the second call, a is familiar with A and B, and both b and c are familiar with A, B, and C. In fact, everybody knows

this. And since we (and, by assumption, the agents) are not interested in the *value* of the secrets (a is familiar with B and hence knows its value, but d does not), we conclude that all agents know the same! Similar reasoning can be done for the other call types with privacy level $\bigcirc$.

b Second, assume the call type is $(\bullet, \diamond, \mathsf{o})$. Suppose that there are exactly three agents a, b and c, and that the following calls just happened: ac, bc and ab (in that order). After the first two calls, a knows A, C while b knows A, B, C. After the ab call, both a and b therefore know A, B, C, i.e., they are experts.

Now, we can show the difference between $\mathsf{o} = \beta$ and $\mathsf{o} = \alpha$. If $\mathsf{o} = \beta$, then in the call ab agent a learns exactly which secrets b held before the call, i.e., the secrets A, B and C. This allows a to deduce that there must have been a call between b and c, where b learned A and C. In that call, c must also have learned the secret B. So after the call ab, a knows that, in particular, F_cB.

In contrast, if $\mathsf{o} = \alpha$, then in the call ab agent a only discovers that b holds a secret X if a does not know X before the call. So a only learns that F_bB (which is always true), and not that F_bA and F_bC. As such, a cannot deduce that the call bc happened. In particular, this means that a doesn't know that F_cB.

Since the goal of the agents is to turn everyone into experts, a may be tempted to call c in order to guarantee F_cB. We know that this call ac is unnecessary, because c already knows B, but in this call type agent a does not.

c If there are exactly three agents and $\mathsf{d} = \diamond$, there is no difference between $\bigcirc$ and $\bigcirc\!\!\!\!\bullet$. After all, if a call ab happens with privacy $\bigcirc\!\!\!\!\bullet$, then c notices that some call has taken place, and that they themselves were not involved. The only possibilities for this call are ab and ba, and with direction $\diamond$ those are equivalent. If the direction is $\triangleright$ or $\triangleleft$, the calls ab and ba have different effects, so the difference between $\bigcirc$ and $\bigcirc\!\!\!\!\bullet$ is relevant.

2.2 Language

Assume a finite set of agents Ag and a set of secrets $\mathsf{S} = \{A \mid a \in Ag\}$. For each direction type $\mathsf{d} \in \{\diamond, \triangleright, \triangleleft\}$ we define the language $\mathcal{L}^{\mathsf{d}}$ as

$$\mathcal{L}^{\mathsf{d}}: \quad \varphi \quad ::= \quad F_aS \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid K_a\varphi \mid [\mathsf{c}]\varphi$$

where $a \in Ag$, $S \in \mathsf{S}$ and $\mathsf{c} \in \mathsf{C}^{\mathsf{d}}$. For any call type $\tau = (\mathsf{p}, \mathsf{d}, \mathsf{o})$, with $\mathcal{L}^{\tau}$ we mean $\mathcal{L}^{\mathsf{d}}$. If the clause $[\mathsf{c}]\varphi$ is omitted, the corresponding language is denoted $\mathcal{L}^{\tau}_{[\,]\text{-free}}$. So the latter is the epistemic sublanguage of $\mathcal{L}^{\tau}$, without dynamic operators. We read F_aS as 'agent a is familiar with the secret S' (or 'S belongs to the set of secrets a knows about') and $K_a\varphi$ as 'agent a knows that formula φ is true'. $\hat{K}_a\varphi$ is shorthand for $\neg K_a \neg\varphi$. Finally, $[\mathsf{c}]\varphi$ is read as "φ will be true after the call c." So $\mathcal{L}^{\tau}$ is a dynamic epistemic language where atoms consist of 'being familiar with' statements about secrets. For $\mathsf{Q} \subseteq \mathsf{S}$, $O_a\mathsf{Q}$ is shorthand for $\bigwedge_{D \in \mathsf{Q}} F_aD \wedge \bigwedge_{D' \in \mathsf{S} \setminus \mathsf{Q}} \neg F_aD'$, i.e., agent a is only (or, exactly) aware of the atoms in Q. Similarly, but somewhat more liberally deviating from the meaning of 'only knowing', we let $O_{ab}\mathsf{Q}$ stand for "together,

agents a and b only know the secrets in Q" and this is defined as $\bigwedge_{B\in\mathsf{Q}}(F_aB \vee F_bB) \wedge \bigwedge_{B\notin\mathsf{Q}} \neg(F_aB \vee F_bB)$. We use *root* as an abbreviation for $\bigwedge_{a\in Ag} O_aA$.

The depth $d(\varphi)$ of a formula φ is the number of nested knowledge operators, i.e, $d(F_aB) = 0$, $d(\neg\varphi) = d([\mathsf{c}]\varphi) = d(\varphi)$, $d(\varphi_1 \wedge \varphi_2) = \max(d(\varphi_1), d(\varphi_2))$ and $d(K_a\varphi) = d(\varphi) + 1$. Note that the call operator $[\mathsf{c}]$ does not affect the depth of a formula.

We will now define the notion of *call sequence*, which will feature both in our semantics and in our object language. Given a call type τ, a call sequence $\vec{\mathbf{c}}$ of direction type d is a finite sequence of calls, in symbols $\vec{\mathbf{c}} = (\mathsf{c}_1, \mathsf{c}_2, \ldots, \mathsf{c}_n)$, with $\mathsf{c}_i \in \mathbf{C}^{\mathsf{d}} (1 \leq i \leq n)$. The empty sequence is denoted by ϵ. The set $\mathbf{C}^{\mathsf{d}}$ collects all call sequences of type d. We sometimes write $\mathbf{C}^{\tau}$ for $\mathbf{C}^{\mathsf{d}}$, where $\tau = (\mathsf{p}, \mathsf{d}, \mathsf{o})$. If the direction type or call type is clear from context or not important, we denote the set of call sequences by $\mathbf{C}$.

Given a call sequence $\vec{\mathbf{c}}$ and a call c we denote by $\mathsf{c}.\vec{\mathbf{c}}$ the prepending of $\vec{\mathbf{c}}$ with c, and by $\vec{\mathbf{c}}.\mathsf{c}$ the postpending of $\vec{\mathbf{c}}$ with c. When $\vec{\mathbf{c}} = (\mathsf{c}_1, \mathsf{c}_2, \ldots, \mathsf{c}_n, \ldots)$ the claim $a \in Ag(\vec{\mathbf{c}})$ is shorthand for $a \in Ag(\mathsf{c}_i)$ for some $i \geq 1$. Finally, by $\vec{\mathbf{c}}.\vec{\mathbf{d}}$ we denote the concatenation of two finite sequences, $\vec{\mathbf{c}}$ and $\vec{\mathbf{d}}$. Note that $\mathsf{c}_1, \mathsf{c}_2$ and $\mathsf{c}_1.\mathsf{c}_2$ basically denote the same thing: a call sequence and the concatenation of two sequences. The length $\ell(\vec{\mathbf{c}})$ of $\vec{\mathbf{c}}$ is defined in a straightforward way: $\ell(\epsilon) = 0$ and $\ell(\mathsf{c}.\vec{\mathbf{c}}) = \ell(\vec{\mathbf{c}}.\mathsf{c}) = 1 + \ell(\vec{\mathbf{c}})$.

For a call $\vec{\mathbf{c}}$ and agent a, the a-reduction of $\vec{\mathbf{c}}$, denoted $\vec{\mathbf{c}}|_a$ consists of the calls in $\vec{\mathbf{c}}$ that a is involved in: $\epsilon_{|_a} = \epsilon$ and $(\vec{\mathbf{c}}.\mathsf{c})_{|_a} = \vec{\mathbf{c}}_{|_a}$ if $a \notin Ag(\mathsf{c})$ and $\vec{\mathbf{c}}_{|_a}.\mathsf{c}$ else.

We will write $\vec{\mathbf{c}} = \vec{\mathbf{d}}$ for two call sequences if they are of the same length, and at every index, they contain the same message (where we identify $a\diamond b$ with $b\diamond a$). So for instance, the sequence $a \triangleright b, b \triangleright c$ is different from $a \triangleright b, c \triangleright b$, but $a\diamond b, b\diamond c = a\diamond b, c\diamond b$. For any call sequence $\vec{\mathbf{c}} \in \mathbf{C}$ we define $[\vec{\mathbf{c}}]\varphi$ as an abbreviation:

$$\begin{array}{lcl} [\epsilon]\varphi & = & \varphi \\ [\mathsf{c}.\vec{\mathbf{c}}]\varphi & = & [\mathsf{c}][\vec{\mathbf{c}}]\varphi \end{array}$$

We also use $\ell(\varphi)$ to denote the maximum nesting depth of calls in φ, i.e., $\ell(F_aB) = 0$, $\ell(\neg\varphi) = \ell(K_a\varphi) = \ell(\varphi)$, $\ell(\varphi \wedge \psi) = \max(\ell(\varphi), \ell(\psi))$ and $\ell([\vec{\mathbf{c}}]\varphi) = \ell(\varphi) + \ell(\vec{\mathbf{c}})$.

3 Semantics

3.1 Initial models and gossip models

Before modeling the entire gossip problem, we start by modeling the initial situation, where the agents have not yet started gossiping. In most variants of the gossip problem it is assumed that in this initial situation it is common knowledge that every agent knows only their own secret. But this is not a necessary assumption: we can describe gossip starting in any initial situation. For example, in the initial situation a could be uncertain about whether b is already familiar with secret C before the gossiping starts. Or the gossip could start from a situation where it is common knowledge that a is spying on b, and that therefore a is initially aware of secrets A and B. The only requirements we place on the initial situation is that every agent must be familiar with

their own secret, and that agents must know which secrets they themselves are aware of.

We denote the initial set of secrets familiar to a in world w as $Q_0(a, w)$, and the initial epistemic accessibility relation as R_0. The requirement that a knows their own secret therefore means that $A \in Q_0(a, w)$, and the requirement that agents know which secrets they are aware of means that if $(w_1, w_2) \in R_0(a)$ then $Q_0(a, w_1) = Q_0(a, w_2)$. The initial model is therefore defined as follows.

Definition 3.1. *An initial model is a triple $I = (W, R_0, Q_0)$ where W is a set of worlds, $R_0 : Ag \to 2^{W \times W}$ maps every agent to an equivalence relation and $Q_0 : Ag \times W \to 2^S$ satisfies (i) $A \in Q_0(a, w)$ for every $(a, w) \in Ag \times W$ and (ii) if $(w_1, w_2) \in R_0(a)$ then $Q_0(a, w_1) = Q_0(a, w_2)$.*

When the agents start to gossip from an initial model I, using call type τ, the result is represented by a gossip model $M^\tau(I) = (St, \sim, Q)$. We define each of the tree parts of $M^\tau(I)$ individually.

Definition 3.2. *Let a call type τ and an initial model $I = (W, R_0, Q_0)$ be given. A gossip state with respect to τ and I is a pair $(w, \vec{\mathbf{c}})$ where $w \in W$ and $\vec{\mathbf{c}} \in \mathbf{C}^\tau$.*

The set of gossip states with respect to τ and I is denoted $St^{I,\tau}$. When τ and I are understood we write St for $St^{I,\tau}$.

At any point in time, the current situation can be uniquely identified by (i) where we started and (ii) which calls happened since then. So the current situation is uniquely identified by a gossip state. The set of secrets known by an agent in any gossip state can be defined inductively.

Definition 3.3. *Let a call type τ and an initial model $I = (W, R_0, Q_0)$ be given. The set of secrets known to an agent in gossip state $(w, \vec{\mathbf{c}})$, denoted $Q^{I,\tau}(a, w, \vec{\mathbf{c}})$, is given inductively by*

- $Q^{I,\tau}(a, w, \epsilon) = Q_0(a, w)$,
- $Q^{I,\tau}(a, w, \vec{\mathbf{c}}.b \triangleright c) = \begin{cases} Q^{I,\tau}(a, w, \vec{\mathbf{c}}) \cup Q^{I,\tau}(b, w, \vec{\mathbf{c}}) & \text{if } a = c \\ Q^{I,\tau}(a, w, \vec{\mathbf{c}}) & \text{otherwise} \end{cases}$
- $Q^{I,\tau}(a, w, \vec{\mathbf{c}}.b \triangleleft c) = \begin{cases} Q^{I,\tau}(a, w, \vec{\mathbf{c}}) \cup Q^{I,\tau}(c, w, \vec{\mathbf{c}}) & \text{if } a = b \\ Q^{I,\tau}(a, w, \vec{\mathbf{c}}) & \text{otherwise} \end{cases}$
- $Q^{I,\tau}(a, w, \vec{\mathbf{c}}.b \diamond c) = \begin{cases} Q^{I,\tau}(b, w, \vec{\mathbf{c}}) \cup Q^{I,\tau}(c, w, \vec{\mathbf{c}}) & \text{if } a \in \{b, c\} \\ Q^{I,\tau}(a, w, \vec{\mathbf{c}}) & \text{otherwise} \end{cases}$

When τ and I are understood we write Q for $Q^{I,\tau}$. We also write $Q_a(w, \vec{\mathbf{c}})$ for $Q(a, w, \vec{\mathbf{c}})$.

Note that this definition captures the meaning of the direction type. Depending on it the secrets are either shared between caller and callee ($a \diamond b$), they are pushed from the caller to the callee ($a \triangleright b$), or they are retrieved by the caller from the callee ($a \triangleleft b$).

By now we have defined the set of gossip states, and "lifted" the initial secret distribution Q_0 to a distribution Q for all gossip states. All that is left to do now is to "lift"

the indistinguishability relation R_0 from the initial situation to all gossip states. Unlike St and Q, the relation R depends on all three parameters, so this definition is slightly more complex. We start with the case $\mathsf{p} = \bigcirc$.

Definition 3.4. *Let a call type $\tau = (\bigcirc, \mathsf{d}, \mathsf{o})$ and an initial model $I = (W, R_0, Q_0)$ be given. For $a \in Ag$, the indistinguishability relation $\sim_a^{I,\tau} \subseteq St^{I,\tau} \times St^{I,\tau}$ is given by*

- $(w_1, \epsilon) \sim_a^{I,\tau} (w_2, \epsilon)$ *iff* $(w_1, w_2) \in R_0(a)$,
- $(w_1, \vec{\mathbf{c}}_1.\mathsf{c}_1) \sim_a^{I,\tau} (w_2, \vec{\mathbf{c}}_2.\mathsf{c}_2)$ *iff* $\mathsf{c}_1 = \mathsf{c}_2$, $(w_1, \vec{\mathbf{c}}_1) \sim_a^{I,\tau} (w_2, \vec{\mathbf{c}}_2)$ *and one of the following five conditions holds:*
  1. $a \notin Ag(\mathsf{c}_1)$,
  2. $\mathsf{c}_1 = a \triangleright b$,
  3. $\mathsf{c}_1 = b \triangleleft a$,
  4. $\mathsf{c}_1 \in \{b \triangleright a, a \triangleleft b, a \diamond b, b \diamond a\}$, $\mathsf{o} = \beta$ *and* $Q_b^{I,\tau}(w_1, \vec{\mathbf{c}}_1) = Q_b^{I,\tau}(w_2, \vec{\mathbf{c}}_2)$ *or*
  5. $\mathsf{c}_1 \in \{b \triangleright a, a \triangleleft b, a \diamond b, b \diamond a\}$, $\mathsf{o} = \alpha$ *and* $Q_a^{I,\tau}(w_1, \vec{\mathbf{c}}_1.\mathsf{c}_1) = Q_a^{I,\tau}(w_2, \vec{\mathbf{c}}_2.\mathsf{c}_1)$.

When I and τ are understood we write $\sim_a$ for $\sim_a^{I,\tau}$.

Let us unpack this definition. The first clause is quite simple: in (w_1, ϵ) and (w_2, ϵ) no calls have yet taken place, so these states are indistinguishable iff w_1 and w_2 are indistinguishable in the initial model I. The second clause is more interesting. Firstly, in order for $(w_1, \vec{\mathbf{c}}_1.\mathsf{c}_1)$ and $(w_2, \vec{\mathbf{c}}_2.\mathsf{c}_2)$ to be indistinguishable, it must be the case that $\mathsf{c}_1 = \mathsf{c}_2$. This is because the privacy type is $\bigcirc$, which allows any agent to see exactly which calls are taking place. Secondly, $(w_1, \vec{\mathbf{c}}_1)$ and $(w_2, \vec{\mathbf{c}}_2)$ must be indistinguishable. In other words, the new call $\mathsf{c}_1 = \mathsf{c}_2$ does not make a forget a distinction between two states that they could make previously. Note that these two conditions imply that if $(w_1, \vec{\mathbf{c}}_1) \sim_a (w_2, \vec{\mathbf{c}}_2)$ then $\vec{\mathbf{c}}_1 = \vec{\mathbf{c}}_2$.

Finally, one of the conditions 1–5 must hold. These conditions represent the fact that the call $\mathsf{c}_1 = \mathsf{c}_2$ may give a insight into the set of secrets held by another agent. If this insight differs between the two states, a can use that to distinguish between them. For example, if $\mathsf{c}_1 = \mathsf{c}_2 = a \diamond b$ and in one state b teaches a the secrets B, C in this final call while in the other state b only teaches a secret B, then a can distinguish between $(w_1, \vec{\mathbf{c}}_1.\mathsf{c}_1)$ and $(w_2, \vec{\mathbf{c}}_2.\mathsf{c}_2)$.

If $a \notin Ag(\mathsf{c}_1)$, then a is not involved in the call and therefore observes nothing about the set of secrets held by any agent. Likewise, if the call is $a \triangleright b$ or $b \triangleleft a$, a observes nothing about b's secrets. If the call is one of $\{b \triangleright a, a \triangleleft b, a \diamond b, b \diamond a\}$, however, then a may gain some insight into b's secrets. Under the "before" observance type β, a can see exactly which secrets b held before the call. So if $Q^{I,\tau}(b, w_1, \vec{\mathbf{c}}_1) \neq Q^{I,\tau}(b, w_2, \vec{\mathbf{c}}_2)$, then a can distinguish between the two states. Under the "after" observance type α, a cannot see exactly which secrets b held before the call. But a does know which secrets they were taught during the call. So if $Q^{I,\tau}(a, w_1, \vec{\mathbf{c}}_1.\mathsf{c}_1) \neq Q^{I,\tau}(a, w_2, \vec{\mathbf{c}}_2.\mathsf{c}_1)$ then a can distinguish between the states.

The $\mathsf{p} = \half$ case differs from $\bigcirc$ in a small but important way.

Definition 3.5. *Let a call type $\tau = (◓, \mathsf{d}, \mathsf{o})$ and an initial model $I = (W, R_0, Q_0)$ be given. For $a \in Ag$, the indistinguishability relation $\sim_a^{I,\tau} \subseteq St^{I,\tau} \times St^{I,\tau}$ is given by*

- $(w_1, \epsilon) \sim_a^{I,\tau} (w_2, \epsilon)$ *iff* $(w_1, w_2) \in R_0(a)$,
- $(w_1, \vec{\mathbf{c}}_1.\mathsf{c}_1) \sim_a^{I,\tau} (w_2, \vec{\mathbf{c}}_2.\mathsf{c}_2)$ *iff* $(w_1, \vec{\mathbf{c}}_1) \sim_a^{I,\tau} (w_2, \vec{\mathbf{c}}_2)$ *and one of the following five conditions holds:*
  1. $a \notin Ag(\mathsf{c}_1) \cup Ag(\mathsf{c}_2)$,
  2. $\mathsf{c}_1 = \mathsf{c}_2 = a \triangleright b$,
  3. $\mathsf{c}_1 = \mathsf{c}_2 = b \triangleleft a$,
  4. $\mathsf{c}_1 = \mathsf{c}_2 \in \{b \triangleright a, a \triangleleft b, a \diamond b, b \diamond a\}$, $\mathsf{o} = \beta$ *and* $Q^{I,\tau}(b, w_1, \vec{\mathbf{c}}_1) = Q^{I,\tau}(b, w_2, \vec{\mathbf{c}}_2)$ *or*
  5. $\mathsf{c}_1 = \mathsf{c}_2 \in \{b \triangleright a, a \triangleleft b, a \diamond b, b \diamond a\}$, $\mathsf{o} = \alpha$ *and* $Q^{I,\tau}(a, w_1, \vec{\mathbf{c}}_1.\mathsf{c}_1) = Q^{I,\tau}(a, w_2, \vec{\mathbf{c}}_2.\mathsf{c}_1)$.

When I and τ are understood we write $\sim_a$ for $\sim_a^{I,\tau}$.

With privacy type ◓, indistinguishability of $(w_1, \vec{\mathbf{c}}_1.\mathsf{c}_1)$ and $(w_2, \vec{\mathbf{c}}_2.\mathsf{c}_2)$ no longer requires $\mathsf{c}_1 = \mathsf{c}_2$. After all, in this synchronous privacy type an agent $a \notin Ag(\mathsf{c}_1) \cup Ag(\mathsf{c}_2)$ does not know exactly which call took place, they only know that some call happened and that they were involved in it. The privacy type only affects the information that agents gain about calls they were not involved in, however, so conditions 2–4 remain unchanged between ○ and ◓ except that we now need to specify that $\mathsf{c}_1 = \mathsf{c}_2$.

Indistinguishability for ● is slightly more complex. This is because with the asynchronous privacy type ● an agent may be unable to distinguish between $(w, \vec{\mathbf{c}})$ and $(v, \vec{\mathbf{d}})$ even when $\vec{\mathbf{c}}$ and $\vec{\mathbf{d}}$ have different lengths.

Definition 3.6. *Let a call type $\tau = (●, \mathsf{d}, \mathsf{o})$ and an initial model $I = (W, R_0, Q_0)$ be given. For $a \in Ag$, the indistinguishability relation $\sim_a^{I,\tau} \subseteq St^{I,\tau} \times St^{I,\tau}$ is the smallest equivalence relation such that*

- *if* $(w_1, w_2) \in R_0(a)$ *then* $(w_1, \epsilon) \sim_a^{I,\tau} (w_2, \epsilon)$,
- *if* $(w_1, \vec{\mathbf{c}}_1) \sim_a^{I,\tau} (w_2, \vec{\mathbf{c}}_2)$ *and* $a \notin Ag(\mathsf{c})$ *then* $(w_1, \vec{\mathbf{c}}_1) \sim_a^{I,\tau} (w_2, \vec{\mathbf{c}}_2.\mathsf{c})$,
- *if* $(w_1, \vec{\mathbf{c}}_1) \sim_a^{I,\tau} (w_2, \vec{\mathbf{c}}_2)$ *and one of the following four conditions holds*
  1. $\mathsf{c} = a \triangleright b$,
  2. $\mathsf{c} = b \triangleleft a$,
  3. $\mathsf{c} \in \{b \triangleright a, a \triangleleft b, a \diamond b, b \diamond a\}$, $\mathsf{o} = \beta$ *and* $Q^{I,\tau}(b, w_1, \vec{\mathbf{c}}_1) = Q^{I,\tau}(b, w_2, \vec{\mathbf{c}}_2)$ *or*
  4. $\mathsf{c} \in \{b \triangleright a, a \triangleleft b, a \diamond b, b \diamond a\}$, $\mathsf{o} = \alpha$ *and* $Q^{I,\tau}(a, w_1, \vec{\mathbf{c}}_1.\mathsf{c}_1) = Q^{I,\tau}(a, w_2, \vec{\mathbf{c}}_2.\mathsf{c}_1)$,

  then $(w_1, \vec{\mathbf{c}}_1.\mathsf{c}) \sim_a^{I,\tau} (w_2, \vec{\mathbf{c}}_2.\mathsf{c})$.

When I and τ are understood we write $\sim_a$ for $\sim_a^{I,\tau}$.

When $a \in Ag(\mathsf{c})$, the privacy type is irrelevant so the conditions under which c allows a to distinguish between the two states are the same in $\bullet$ as in $\circ$ and $\circledcirc$. If $a \notin Ag(\mathsf{c})$, however, the privacy type is important. With $\mathsf{p} = \bullet$, agent a is completely unaware of such a call taking place, so $(w_1, \vec{\mathbf{c}}_1) \sim_a (w_2, \vec{\mathbf{c}}_2)$ implies $(w_1, \vec{\mathbf{c}}_1) \sim_a (w_2, \vec{\mathbf{c}}_2.\mathsf{c})$. Because $\sim_a$ is defined to be the smallest equivalence relation closed under these properties, we also immediately obtain that if $a \notin Ag(\vec{\mathbf{d}}_1) \cup Ag(\vec{\mathbf{d}}_2)$ then $(w_1, \vec{\mathbf{c}}_1) \sim_a (w_2, \vec{\mathbf{c}}_2)$ implies $(w_1, \vec{\mathbf{c}}_1.\vec{\mathbf{d}}_1) \sim_1 (w_2, \vec{\mathbf{c}}_1.\vec{\mathbf{d}}_1)$.

Now that we have defined the set St of states, the indistinguishability relations $\sim_a$ and the sets Q of known secrets, defining a gossip model is simply a matter of combining these three elements.

Definition 3.7. *Let a call type τ and an initial model $I = (W, R_0, Q_0)$ be given. The gossip model with respect to τ and I, denoted $M^\tau(I)$, is given by $M^\tau(I) = (St^{I,\tau}, \sim^{I,\tau}_{a \in Ag}, Q^{I,\tau})$.*

The class of all gossip models with respect to τ is denoted $\mathcal{M}^\tau$.

3.2 The tree model

One can coherently describe the gossip problem starting in any initial situation. Typically, however, only one initial situation is considered, namely the one where it is common knowledge that every agent knows only their own secret. In addition to modeling the gossip problem in general, we would therefore also like to represent the gossip problem starting in that specific initial situation. In general, every gossip model can be seen as a forest, with the initial model forming its roots. In the special case where it is common knowledge that every agent knows only their own secret we require only one world in our initial model, so the resulting gossip model is a tree. We therefore refer to it as the *tree model*.

Definition 3.8. *Let $I_{tree} = (W_{tree}, R_{tree}, O_{tree})$ be the initial model given by $W_{tree} = \{w_{root}\}$, $R_{tree}(a) = W_{tree} \times W_{tree}$ for every a and $O_{tree}(a, w_{root}) = \{A\}$ for every a. The gossip model $M^\tau_{tree} := M^\tau(I_{tree})$ is called the tree model for call type τ.*

Note that the tree model is a specific tree; there are other gossip models that are trees, but we do not refer to them as the tree model.

3.3 Semantics

Given a gossip model, the semantics for $\mathcal{L}^\tau$ are pretty straightforward.

Definition 3.9. *The satisfaction relation $\models$ between gossip states and formulas is defined inductively as follows.*

$$
\begin{array}{lcl}
M, (w, \vec{\mathbf{c}}) \models F_a B & \Leftrightarrow & B \in Q_a(w, \vec{\mathbf{c}}) \\
M, (w, \vec{\mathbf{c}}) \models \neg\varphi & \Leftrightarrow & M, (w, \vec{\mathbf{c}}) \not\models \varphi \\
M, (w, \vec{\mathbf{c}}) \models \varphi \wedge \psi & \Leftrightarrow & M, (w, \vec{\mathbf{c}}) \models \varphi \text{ and } M, (w, \vec{\mathbf{c}}) \models \psi \\
M, (w, \vec{\mathbf{c}}) \models K_a\varphi & \Leftrightarrow & M, (w', \vec{\mathbf{c}}') \models \varphi \text{ for all } (w', \vec{\mathbf{c}}') \text{ s. t. } (w, \vec{\mathbf{c}}) \sim_a (w', \vec{\mathbf{c}}') \\
M, (w, \vec{\mathbf{c}}) \models [\mathsf{c}]\varphi & \Leftrightarrow & M, (w, \vec{\mathbf{c}}.\mathsf{c}) \models \varphi
\end{array}
$$

If $M, (w, \vec{\mathbf{c}}) \models \varphi$ for every gossip state $(w, \vec{\mathbf{c}})$ of M, we say that φ is valid on M and write $M \models \varphi$. If $M \models \varphi$ for all $M \in \mathcal{M}^\tau$ we say that φ is valid and write $\models^\tau \varphi$. If $M^\tau_{tree} \models \varphi$ we say that φ is valid in the tree model and write $\models^\tau_{tree} \varphi$.

We often omit reference to the model M where that should not cause confusion, and write $w, \vec{\mathbf{c}} \models \varphi$ for $M, (w, \vec{\mathbf{c}}) \models \varphi$.

We can now discuss a the properties of various kinds of gossip. First, let us consider a number of basic properties shared by all call types. Recall that $O_a\mathsf{Q}$, where $\mathsf{Q} \subseteq \mathsf{S}$, means that a knows exactly the set Q of secrets.

Proposition 3.10. *For every call type τ, every $a \in Ag$, every $B, D \in \mathsf{S}$, every $\vec{\mathbf{c}} \in \mathbf{C}^\tau$, every $\mathsf{c}_1, \ldots, \mathsf{c}_n \in \mathsf{C}^\tau$, every $\varphi \in \mathcal{L}^\tau$ and every world w the following properties hold.*

Unique	$\models^\tau \bigvee_{\mathsf{Q} \subseteq \mathsf{S}} (O_a\mathsf{Q} \wedge \bigwedge_{\mathsf{Q}' \neq \mathsf{Q}} \neg O_a\mathsf{Q}')$
Own	$\models^\tau F_a A$
Persistence	$\models^\tau F_a B \to [\mathsf{c}]F_a B$
Composition	$w, \vec{\mathbf{c}}.(\mathsf{c}_1, \mathsf{c}_2, \ldots, \mathsf{c}_n) \models^\tau \varphi$ *iff* $w, \vec{\mathbf{c}} \models^\tau [\mathsf{c}_1][\mathsf{c}_2] \ldots, [\mathsf{c}_n]\varphi$
F-Introspection	$\models^\tau (F_a D \leftrightarrow K_a F_a D) \wedge (\neg F_a D \leftrightarrow K_a \neg F_a D)$

Except for **F-introspection**, the above properties follow immediately from the semantics. To see why **F-introspection** is valid, note that the initial model is defined in such a way that if two worlds are a-indistinguishable, then a is aware of the same secrets in both worlds. This property is preserved when calls happen, so an agent a always knows which secrets they are aware of.

The property **Unique** states that there is always exactly one set of secrets that agent a is aware of, and **Own** guarantees that this set includes A. **Persistence** tells us that if an agents is familiar with a secret, this will remain true in the future. According to **Composition**, the definition of composition behaves as desired, and **F-Introspection** tells us that agents know which secrets they are familiar with.

The 10 properties described in the following proposition do depend on the privacy type. Some of them also depend on whether we consider only the tree model, or all gossip models.

Proposition 3.11. *Consider the following schemes, where a, b and c are different agents. (Recall that $\hat{K}_a = \neg K_a \neg$ and root $= \bigwedge_{a \in Ag} O_a A$.)*

fi	*Full Ignorance*	$\hat{K}_a F_b C$	
pp	*Postponed Ignorance*	$[\mathsf{c}]\hat{K}_a F_b C$	*where* $a \notin Ag(\mathsf{c})$
vi	*Visibility*	$[\mathsf{c}]K_a(F_b C \vee F_c B)$	*where* $Ag(\mathsf{c}) = (b, c)$
co	*Commute*	$[\mathsf{c}][\mathsf{d}]\varphi \leftrightarrow [\mathsf{d}][\mathsf{c}]\varphi$,	*where* $Ag(\mathsf{c}) \cap Ag(\mathsf{d}) = \emptyset$
fk	*Full Knowledge*	$\varphi \leftrightarrow K_a\varphi$	
if	*Initial Full Knowledge*	$root \to (\varphi \leftrightarrow K_a\varphi)$	
fp	*Full Privacy*	$K_a\varphi \leftrightarrow [\mathsf{c}]K_a\varphi$	*where* $a \notin Ag(\mathsf{c})$
pr	*Perfect Recall*	$K_a[\mathsf{c}]\varphi \to [\mathsf{c}]K_a\varphi$	*where* $a \notin Ag(\mathsf{c})$
or	*Own Perfect Recall*	$K_a[\mathsf{c}]\varphi \to [\mathsf{c}]K_a\varphi$	*where* $a \in Ag(\mathsf{c})$
ns	*No Surprises*	$[\mathsf{c}]K_a\varphi \to K_a[\mathsf{c}]\varphi$	

Then the following table shows the effect of the different privacy levels on those schemes as a validity:

$\tau(\mathsf{p})$	*fi*	*pp*	*vi*	*co*	*fk*	*if*	*fp*	*pr*	*or*	*ns*
○	×	×	✓	✓	×	×	×	✓	✓	×
◓	×	✓	×	×	×	×	×	×	✓	×
●	✓	✓	×	✓	×	×	✓	×	×	×
○($tree$)	×	×	✓	✓	✓	✓	×	✓	✓	✓
◓($tree$)	×	✓	×	×	×	✓	×	×	✓	×
●($tree$)	✓	✓	×	✓	×	×	✓	×	×	×

The properties *fi*, *pp* and *vi* very directly describe what agents know about calls that they are not involved in. If $\tau(\mathsf{p}) = \bullet$, agent a can never be certain that c hasn't told b their secret, so *fi* holds. If $\tau(\mathsf{p}) = ◓$ then a may initially be certain that c hasn't told b their secret, but any call where a is not involved might have been between b and c, and therefore makes a uncertain about F_bC. As a result, *pp* holds. Finally, if $\tau(\mathsf{p}) = ○$, then a knows exactly which calls happen, so after a call between b and c agent a will, depending on the call direction, know either F_bC or F_cB. So *vi* holds.

The property *co* is interesting because it holds for ○ and ●, but for different reasons. If $\tau(\mathsf{p}) = \bullet$, the calls c and d commute because no agent can distinguish between the two orders. If $\tau(\mathsf{p}) = ○$, then c and d commute because it is common knowledge that exactly those two calls happened and that their order does not matter.[3] If $\tau(\mathsf{p}) = ◓$, the two calls do not commute because the agents cannot be certain that $Ag(\mathsf{c}) \cap Ag(\mathsf{d}) = \emptyset$. For example, $(w_{root}, \epsilon) \models [c \diamond d][a \diamond b]K_a \neg F_c A$ but $(w_{root}, \epsilon) \not\models [a \diamond b][c \diamond d]K_a \neg F_c A$, because as far as a can tell the second call could have been $b \diamond c$ instead of $c \diamond d$.

Now, consider the properties *fk* and *if*. Any gossip state in the tree model is of the form $(w_{root}, \vec{\mathbf{c}})$ for some $\vec{\mathbf{c}}$. So if an agent a can identify the exact call sequence $\vec{\mathbf{c}}$, that agent knows exactly which state they are in, so we have $\varphi \leftrightarrow K_a\varphi$. With ○, all agents always know the call sequence, so $\models_{tree}^{(○,\mathsf{d},\mathsf{o})} fk$. With ◓, agents know the call sequence if it happens to be ϵ, so $(w_{root}, \epsilon) \models^{(◓,\mathsf{d},\mathsf{o})} \varphi \leftrightarrow K_a\varphi$. Since (w_{root}, ϵ) is uniquely (in the tree model) identified by the formula $root$, it follows that $\models_{tree}^{(◓,\mathsf{d},\mathsf{o})} if$. With ● the agents never know the exact call sequence, so neither *fk* nor *if* are valid for that privacy type.

The validity of *fp* for ●, like that of *fi*, follows immediately from thew fact that a cannot tell whether c has happened unless $a \in Ag(\mathsf{c})$. The property *pr* holds for ○ because in that privacy type agent a knows which call has taken place. This allows a to turn their pre-call hypothetical reasoning $K_a[\mathsf{c}]\varphi$ into post-call unconditional reasoning $[\mathsf{c}]K_a\varphi$. With privacy ◓ this doesn't work, because while a can do hypothetical reasoning about what would happen if c were to place, after the call they will not be certain that c was indeed the call that took place. If a themselves is involved in c then they can be certain that c took place, so *or* does hold for ◓. For ● even *or* fails, since after c happened a cannot be certain that only a took place; a cannot distinguish between c and $\mathsf{c.d}$. Finally, *ns* holds only if a knows in advance what the result of a call c will be. This is only valid in the tree model with privacy ○.

Note that for each privacy type we can find a validity that holds for that type but

[3] This can be shown quite easily by induction on the depth of φ.

not for either of the other types. We have $\models^\tau vi$ if and only if $\tau(\mathsf{p}) = \circ$, $\models^\tau pp \wedge or$ if and only if $\tau(\mathsf{p}) = \mathbin{\rotatebox{180}{$\bullet$}}$ and $\models^\tau fi$ if and only if $\tau(\mathsf{p}) = \bullet$.

The other parameters can also be uniquely identified by a validity.

Proposition 3.12. *For any three different agents a, b and c and their associated secrets A, B and C the following equivalences hold.*

$$\begin{array}{lll}
\models^\tau (\neg F_a B \wedge \neg F_b A) \to [ab](\neg F_a B \wedge F_b A) & \textit{iff} & \tau(\mathsf{d}) = \triangleright \\
\models^\tau (\neg F_a B \wedge \neg F_b A) \to [ab](F_a B \wedge \neg F_b A) & \textit{iff} & \tau(\mathsf{d}) = \triangleleft \\
\models^\tau (\neg F_a B \wedge \neg F_b A) \to [ab](F_a B \wedge F_b A) & \textit{iff} & \tau(\mathsf{d}) = \diamond \\
\models^\tau (F_b C \wedge F_a C) \to ([ab]K_a F_b C \vee [ab]K_b F_a C) & \textit{iff} & \tau(\mathsf{o}) = \beta \\
\models^\tau (K_a((O_b ABC \wedge O_c ABC) \vee (O_b B \wedge O_c AC)) \wedge \neg K_a F_b C \wedge F_a C) & & \\
\qquad \to ([ab]\neg K_a F_c B \wedge [ba]\neg K_a F_c B) & \textit{iff} & \tau(\mathsf{o}) = \alpha
\end{array}$$

The validities for $\triangleright$, $\triangleleft$ and $\diamond$ are simple, so we don't discuss them further. With regard to the validity for β, recall that observance level β means that if a is the listener in a call ab then a inspects b's secrets before merging them with their own set of secrets. In particular, this means that a will become aware of exactly which secrets b knew at the time of the call. Regardless of the call direction a will be the listener in at least one of the calls ab and ba, so it follows that the fourth formula is valid if $\tau(\mathsf{o}) = \beta$.

The validity characterizing α is slightly more complicated. Let us first look at the situation described by the fifth formula's antecedent. It states that a knows secret C and is uncertain between exactly two possibilities for the sets of secrets known by b and c: either b knows only B and c knows AC, or b and c both know ABC. This situation typically occurs with call type $(\bullet, \diamond, \alpha)$ where exactly one call ac has taken place, but it can happen for any call type given the appropriate initial model. In this situation, if a were to learn that b already knows secret C, they would also learn that c knows B. However, if the observance level is α, then in any call the listener will not observe the informer's secrets directly; instead the listener only observes their new set of secrets after the informer's secrets have been added. In particular, if the listener already knows a particular secret then they will not find out whether the informer also knew that secret. As a result, under observance α neither the call ab nor the call ba will inform a about whether b knew A or C before the call. Agent a's uncertainty about whether c knows B is therefore not removed by these calls, so the fifth formula of Proposition 3.12 is valid if $\tau(\mathsf{o}) = \alpha$.

3.4 n-Bisimilarity

Every gossip model is, in particular, a model of modal logic. We can therefore apply some of the methods of modal logic. In particular, we will make use of n-bisimilarity.

Definition 3.13 (n-bisimulation)**.** *Let $n \in \mathbb{N}$, and let $M = (St, \sim, Q)$ and $M' = (St', \sim', Q')$ be two gossip models. The states $s \in St$ and $s' \in St'$ are n-bisimilar, denoted $M, s \leftrightsquigarrow_n M', s'$ if*

Atoms *for every $a \in Ag$, $Q_a(s) = Q'_a(s')$,*

Forth *if $n > 0$, then for every $t \in St$ and $a \in Ag$, if $s \sim_a t$ then for some $t' \in St'$, also $s' \sim_a t'$ and $M, t \leftrightsquigarrow_{n-1} M', t'$,*

Back *if $n > 0$, then for every $t' \in St'$ and $a \in Ag$, if $s' \sim_a t'$ then for some $t \in St$, also $s \sim_a t$ and $M, t \leftrightsquigarrow_{n-1} M', t'$.*

When the model is clear from context we write $s \leftrightsquigarrow_n s'$ for $M, s \leftrightsquigarrow_n M', s'$. The following theorem is well known in modal logic, see for example [11]:

Theorem 3.14. *Let M, s and M', s' be pointed models. Then the following statements are equivalent:*

1. *M, s and M', s' are n-bisimilar,*
2. *for every $\varphi \in \mathcal{L}^{\tau}_{[\,]\text{-free}}$ of depth at most n, $M, s \models \varphi$ iff $M', s' \models \varphi$.*

It is easy to see that the relation $\leftrightsquigarrow_n$ is an equivalence relation. Furthermore, since our set of atoms is finite, for given n the number of different equivalence classes with respect to $\leftrightsquigarrow_n$ is finite.

The following theorem will be important at several places in the paper, and states that n-bisimilarity is preserved under call sequences.

Theorem 3.15 (Preservation of $\leftrightsquigarrow_n$)**.** *If $(w_1, \vec{\mathbf{c}}_1) \leftrightsquigarrow_n (w_2, \vec{\mathbf{c}}_2)$ then for every call sequence $\vec{\mathbf{d}}$, $(w_1, \vec{\mathbf{c}}_1.\vec{\mathbf{d}}) \leftrightsquigarrow_n (w_2, \vec{\mathbf{c}}_2.\vec{\mathbf{d}})$.*

Proof. We give the proof for $\tau(\mathsf{p}) = \bullet$, the proofs for $\tau(\mathsf{p}) = \circ$ and $\tau(\mathsf{p}) = \boldsymbol{\ominus}$ are similar but simpler.

It suffices to show that the theorem holds whenever $\vec{\mathbf{d}}$ is a single call ab, the theorem then follows by repeatedly adding single calls. We use a proof by induction on n.

First, note that the sets of secrets known to the agents after a call are fully determined by the sets of secrets known to the agents before the call. Since $(w_1, \vec{\mathbf{c}}_1) \leftrightsquigarrow_n (w_2, \vec{\mathbf{c}}_2)$ it follows that, in particular, the two states agree on all atoms. It follows that $(w_1, \vec{\mathbf{c}}_1.\vec{\mathbf{d}})$ and $(w_2, \vec{\mathbf{c}}_2.\vec{\mathbf{d}})$ also agree on all atoms. So **Atoms** holds for $(w_1, \vec{\mathbf{c}}_1.\vec{\mathbf{d}})$ and $(w_2, \vec{\mathbf{c}}_2.\vec{\mathbf{d}})$.

As base case, suppose that $n = 0$. In that case, **Forth** and **Back** are trivial. We already showed that **Atoms** holds, so $(w_1, \vec{\mathbf{c}}_1.ab) \leftrightsquigarrow_0 (w_2, \vec{\mathbf{c}}_2.ab)$. Suppose then as induction hypothesis that $n > 0$ and that the theorem holds for all $n' < n$. We have already shown that **Atoms** holds, left to show is that **Forth** and **Back** hold.

Take any agent c, and any state $(w_1', \vec{\mathbf{c}}_1') \sim_c (w_1, \vec{\mathbf{c}}_1.ab)$. First, suppose that $c \notin \{a, b\}$. Then any call sequence is c-indistinguishable from $\vec{\mathbf{c}}_1.ab$ if and only if it is c-indistinguishable from $\vec{\mathbf{c}}_1$. So we have $(w_1', \vec{\mathbf{c}}_1') \sim_c (w_1, \vec{\mathbf{c}}_1)$. By the n-bisimilarity of $(w_1, \vec{\mathbf{c}}_1)$ and $(w_2, \vec{\mathbf{c}}_2)$, it follows that there is $(w_2', \vec{\mathbf{c}}_2') \sim_c (w_2, \vec{\mathbf{c}}_2)$ such that $(w_1', \vec{\mathbf{c}}_1') \leftrightsquigarrow_{n-1} (w_2', \vec{\mathbf{c}}_2')$. Because $c \notin \{a, b\}$, it follows that $(w_2', \vec{\mathbf{c}}_2') \sim_c (w_2, \vec{\mathbf{c}}_2.ab)$ and $(w_1', \vec{\mathbf{c}}_1') \leftrightsquigarrow_{n-1} (w_2', \vec{\mathbf{c}}_2')$. We have shown that **Forth** holds for $c \notin \{a, b\}$.

Suppose then that $c \in \{a, b\}$. The c-reductions of $\vec{\mathbf{c}}_1, ab$ and $\vec{\mathbf{c}}_1'$ must be the same, so $\vec{\mathbf{c}}_1'$ is of the form $\vec{\mathbf{c}}_1' = \vec{\mathbf{d}}_1.ab.\vec{\mathbf{d}}_1'$, where c doesn't occur in $\vec{\mathbf{d}}_1'$ and $(w_1, \vec{\mathbf{c}}_1) \sim_c (w_1', \vec{\mathbf{d}}_1)$. By the assumption that $(w_1, \vec{\mathbf{c}}_1) \leftrightsquigarrow_n (w_2, \vec{\mathbf{c}}_2)$ there is $(w_2', \vec{\mathbf{d}}_2) \sim_c (w_2, \vec{\mathbf{c}}_2)$ such that $(w_2', \vec{\mathbf{d}}_2) \leftrightsquigarrow_{n-1} (w_1', \vec{\mathbf{d}}_1)$. Now, consider the state $(w_2', \vec{\mathbf{d}}_2.ab.\vec{\mathbf{d}}_1')$. By the induction hypothesis, we have $(w_1', \vec{\mathbf{d}}_1.ab.\vec{\mathbf{d}}_1') \leftrightsquigarrow_{n-1} (w_2', \vec{\mathbf{d}}_2.ab.\vec{\mathbf{d}}_1')$.

Furthermore, consider $(w_2, \vec{\mathbf{c}}_2.ab)$ and $(w_2', \vec{\mathbf{d}}_2.ab.\vec{\mathbf{d}}_1')$. Since c doesn't occur in $\vec{\mathbf{d}}_1'$ and $(w_2, \vec{\mathbf{c}}_2) \sim_c (w_2', \vec{\mathbf{d}}_2)$, c cannot distinguish between $(w_2, \vec{\mathbf{c}}_2.ab)$ and $(w_2', \vec{\mathbf{d}}_2.ab.\vec{\mathbf{d}}_1')$ unless the information gained in the ab call differs between the two sequences. But the information gained by c from the ab call is the same in $(w_2, \vec{\mathbf{c}}_2.ab)$ and $(w_1, \vec{\mathbf{c}}_1.ab)$, since $(w_2, \vec{\mathbf{c}}_2) \leftrightsquigarrow_n (w_2, \vec{\mathbf{c}}_1)$, so they agree on all atoms. Similarly, the information c learns from the ab call is the same in $(w_1, \vec{\mathbf{c}}_1.ab)$ and $(w_1', \vec{\mathbf{d}}_1.ab.\vec{\mathbf{d}}_1')$, because those two states are a-indistinguishable. Finally, c learns the same in $(w_1', \vec{\mathbf{d}}_1.ab.\vec{\mathbf{d}}_1')$ and $(w_2', \vec{\mathbf{d}}_2.ab.\vec{\mathbf{d}}_1')$ because $(w_2', \vec{\mathbf{d}}_2) \leftrightsquigarrow_{n-1} (w_1', \vec{\mathbf{d}}_1)$. Taken together, these three facts imply that c learns the same in the ab calls of $(w_2, \vec{\mathbf{c}}_2.ab)$ and $(w_2', \vec{\mathbf{d}}_2.ab.\vec{\mathbf{d}}_1')$, so $(w_2, \vec{\mathbf{c}}_2.ab) \sim_c (w_2', \vec{\mathbf{d}}_2.ab.\vec{\mathbf{d}}_1')$.

We have now shown that $(w_1', \vec{\mathbf{d}}_1.ab.\vec{\mathbf{d}}_1') \leftrightsquigarrow_{n-1} (w_2', \vec{\mathbf{d}}_2.ab.\vec{\mathbf{d}}_1')$ and $(w_2, \vec{\mathbf{c}}_2.ab) \sim_c (w_2', \vec{\mathbf{d}}_2.ab.\vec{\mathbf{d}}_1')$, so we have shown that **Forth** holds for $c \in \{a, b\}$. We had already shown that **Forth** holds for $c \notin \{a, b\}$.

Due to symmetry it can be shown in the same way that **Back** holds. So this concludes the induction step and thereby the proof. □

It follows truth is preserved under n-bisimilarity for every formula of depth n, not just for epistemic formulas.

Corollary 3.16. *For every $\varphi \in \mathcal{L}^\tau$, if $d(\varphi) = n$ and $(w_1, \vec{\mathbf{c}}_1) \leftrightsquigarrow_n (w_2, \vec{\mathbf{c}}_2)$ then $(w_1, \vec{\mathbf{c}}_1) \models \varphi$ if and only if $(w_2, \vec{\mathbf{c}}_2) \models \varphi$.*

Furthermore, we can use preservation of n-bisimilarity to prove a bound on the length of call sequences that we need to consider.

Proposition 3.17. *Let a call type τ and a formula φ be given, and let $n = d(\varphi)$. For any gossip model M and any state $(w, \vec{\mathbf{d}}.\vec{\mathbf{c}})$ of that model, if $(w, \vec{\mathbf{d}}.\vec{\mathbf{c}}) \models \varphi$ then there is a call sequence $\vec{\mathbf{c}}'$ such that $\ell(\vec{\mathbf{c}}') \leq f(n)$ and $(w, \vec{\mathbf{d}}.\vec{\mathbf{c}}') \models \varphi$, where $f(n)$ is the number of n-bisimilarity classes.*

Proof. If $\ell(\vec{\mathbf{c}}) \leq f(n)$, the proposition is trivial. So suppose that $\ell(\vec{\mathbf{c}}) > f(n)$. Then there must be two different initial fragments $\vec{\mathbf{c}}_1$ and $\vec{\mathbf{c}}_2$ of $\vec{\mathbf{c}}$ such that $(w, \vec{\mathbf{d}}.\vec{\mathbf{c}}_1) \leftrightsquigarrow_n (w, \vec{\mathbf{d}}.\vec{\mathbf{c}}_2)$. Now, let $\vec{\mathbf{c}}_3$ be the remainder of $\vec{\mathbf{c}}$ after $\vec{\mathbf{c}}_2$, i.e., $\vec{\mathbf{c}} = \vec{\mathbf{c}}_2.\vec{\mathbf{c}}_3$. By Theorem 3.15, $(w, \vec{\mathbf{d}}.\vec{\mathbf{c}}_1.\vec{\mathbf{c}}_3) \leftrightsquigarrow_n (w, \vec{\mathbf{d}}.\vec{\mathbf{c}}_2.\vec{\mathbf{c}}_3)$. Since $\vec{\mathbf{c}}_2.\vec{\mathbf{c}}_3 = \vec{\mathbf{c}}$ and truth of φ is preserved under n-bisimulation, it follows that $(w, \vec{\mathbf{d}}.\vec{\mathbf{c}}_1.\vec{\mathbf{c}}_3) \models \varphi$. We have $\ell(\vec{\mathbf{c}}_1.\vec{\mathbf{c}}_3) < \ell(\vec{\mathbf{c}})$, so $\vec{\mathbf{c}}$ is not the shortest call sequences after which φ is true. We can use this procedure to find progressively shorter call sequences after which φ is true, until we find one that is of length at most $f(n)$. □

3.5 Decidability of model checking and validity checking

For any logic, there are two main computational problems that we would like to solve, namely the model checking problem and the validity checking problem. Validity checking can be defined as usual, except that we can define two different versions: one for $\models$ and one for $\models_{tree}$.

Definition 3.18. *The validity checking problem for a call type τ is to determine, for any input formula $\varphi \in \mathcal{L}^\tau$, whether $\models^\tau \varphi$. The tree-validity checking problem for a call type τ is to determine, for any input formula $\varphi \in \mathcal{L}^\tau$, whether $\models^\tau_{tree} \varphi$.*

When defining the model checking problem, we have to be a bit more careful. The input for a computational problem is generally assumed to be finite, but every gossip model is infinite. It therefore makes more sense to define the model checking problem in terms of an initial model, as opposed to a gossip model.

Definition 3.19. *The model checking problem for a call type τ is to determine, for any input formula $\varphi \in \mathcal{L}^\tau$, any finite initial model $I = (W, R, Q)$, any $w \in W$ and any call sequence $\vec{\mathbf{c}}$, whether $M(I), (w, \vec{\mathbf{c}}) \models \varphi$.*

In [4], it was shown that model checking for the tree model with call type $\tau = (\bullet, \mathsf{d}, \alpha)$ is decidable, when restricted to formulas of depth at most 1. Here, we will show that model checking, validity checking and tree-validity checking are all decidable for every call type and for unrestricted formula depth.

Theorem 3.20. *For any call type τ, the model checking problem, validity problem and tree-validity problem for τ are decidable.*

Proof. We start with the model checking problem. It is not possible to fully construct the model $M(I)$, since that model is infinite. What we can do, however, is construct $M(I)$ lazily, i.e., we only construct a state $(w, \vec{\mathbf{c}})$ once we have to do so in order to evaluate a formula.

In the synchronous case, so if $\tau(\mathsf{p}) \in \{\circ, \bullet\}$, indistinguishability is only possible for call sequences of the same length. So $(w, \vec{\mathbf{c}}) \sim_a (w', \vec{\mathbf{c}}')$ implies $\ell(\vec{\mathbf{c}}) = \ell(\vec{\mathbf{c}}')$.

When evaluating $[\vec{\mathbf{d}}]\varphi$ in $(w, \vec{\mathbf{c}})$, we need to create the state $(w, \vec{\mathbf{c}}.\vec{\mathbf{d}})$ and evaluate φ there. When evaluating $K_a\varphi$ in $(w, \vec{\mathbf{c}})$ we need to create every state $(w', \vec{\mathbf{c}}')$ such that $(w, \vec{\mathbf{c}}) \sim_a (w', \vec{\mathbf{c}}')$, and evaluate φ there. Overall, in order to determine whether $M(I), (w, \vec{\mathbf{c}}) \models \varphi$ we only need to create states $(w', \vec{\mathbf{c}}')$ where $\ell(\vec{\mathbf{c}}') \leq \ell(\vec{\mathbf{c}}) + \ell(\varphi)$. It follows that model checking for $\circ$ and $\bullet$ is decidable.

In the asynchronous case, things are more complicated. When evaluating $[\vec{\mathbf{d}}]\varphi$ in $(w, \vec{\mathbf{c}})$, we still only need to create the state $(w, \vec{\mathbf{c}}.\vec{\mathbf{d}})$ and evaluate φ there. When evaluating $K_a\varphi$, however, we would need to create every state $(w', \vec{\mathbf{c}}') \sim_a (w, \vec{\mathbf{c}})$, and there are infinitely many such states.

Fortunately, when $(w, \vec{\mathbf{c}}) \sim_a (w', \vec{\mathbf{d}})$, $(w, \vec{\mathbf{c}}) \sim_a (w', \vec{\mathbf{d}}')$ and $(w', \vec{\mathbf{d}}) \leftrightsquigarrow_{n-1} (w', \vec{\mathbf{d}}')$, invariance of φ under n-bisimulation implies that we only need to check φ in only one of the states.

Take any $(w', \vec{\mathbf{d}})$ such that $(w, \vec{\mathbf{c}}) \sim_a (w', \vec{\mathbf{c}}')$. Then $\vec{\mathbf{c}}$ and $\vec{\mathbf{c}}'$ have the same a-reductions, so $\vec{\mathbf{c}} = \vec{\mathbf{c}}_0.\mathsf{c}_1.\vec{\mathbf{c}}_1.\cdots.\mathsf{c}_m.\vec{\mathbf{c}}_m$ and $\vec{\mathbf{d}} = \vec{\mathbf{d}}_0.\mathsf{c}_1.\vec{\mathbf{d}}_1.\cdots.\mathsf{c}_m.\vec{\mathbf{d}}_m$, where a does not occur in any $\vec{\mathbf{c}}_i$ or $\vec{\mathbf{d}}_i$.

Using Theorem 3.15, we can replace every $\vec{\mathbf{d}}_i$ by $\vec{\mathbf{d}}'_i$ such that (i) $\ell(\vec{\mathbf{d}}_i) \leq f(n-1)$ and (ii) for every i, $(w', \vec{\mathbf{d}}_0.\mathsf{c}_0.\cdots.\vec{\mathbf{d}}_i) \leftrightsquigarrow_n (w', \vec{\mathbf{d}}'_0.\mathsf{c}_1.\cdots.\vec{\mathbf{d}}'_i)$. Let $\vec{\mathbf{d}}' = \vec{\mathbf{d}}'_0.\mathsf{c}_0.\cdots.\vec{\mathbf{d}}_m$. Then at every call c_i, the set of secrets observed by a is the same in $\vec{\mathbf{d}}$ and $\vec{\mathbf{d}}'$. It follows that $(w', \vec{\mathbf{d}}) \sim_a (w', \vec{\mathbf{d}}')$. By assumption, $(w', \vec{\mathbf{d}}) \sim_a (w, \vec{\mathbf{c}})$, so because $\sim_a$ is an equivalence relation it follows that $(w', \vec{\mathbf{d}}') \sim_a (w, \vec{\mathbf{c}})$.

The length of $\vec{\mathbf{d}}'$ is bounded by $m + f(n-1) \cdot (m+1)$. So in order to determine whether $M(I), (w, \vec{\mathbf{c}}) \models K_a\varphi$ we only have to check whether $M(I), (w', \vec{\mathbf{d}}') \models \varphi$ for all $(w', \vec{\mathbf{d}}') \sim_a (w, \vec{\mathbf{c}})$ where the length of $\vec{\mathbf{d}}'$ is bounded. The model checking problem for $\bullet$ is therefore decidable.

The fact that the model checking problem is decidable for every call type can now be used to show that the validity and tree validity checking problems are decidable. Tree-validity can be reduced to checking whether $M(I_{tree}), (w_{root}, \vec{\mathbf{c}}) \models \varphi$ for all $\vec{\mathbf{c}}$ of length at most $f(n)$. Validity can be reduced to checking whether $M(I), (w, \vec{\mathbf{c}}) \models \varphi$ for some maximal set of non-n-bisimilar initial models I, all worlds of those models and all $\vec{\mathbf{c}}$ of length at most $f(n)$. □

4 Axiomatisation: Synchronous Case

We have just shown that the validity problem is decidable, so the existence of a recursive axiomatization is trivial; we could simply use one axiom

$$\textbf{All} \qquad \vdash \varphi \qquad \text{where } \models \varphi.$$

That axiomatization is neither very elegant nor very insightful, however, so we present some more concrete axiomatizations as well.

For every call type τ, we will present two axiomatizations: one that is sound and complete for τ on all gossip models, and one that is sound and complete for τ on the tree model. We start by presenting a proof system that is sound and complete for call-free formulas on all gossip models. Later on, we will add axioms for calls, and rules that renders the proof system complete for the tree model.

Propositional		**Knowledge**	
Prop	propositional tautologies	**K**	$K_a(\varphi \to \psi) \to (K_a\varphi \to K_a\psi)$
MP	$\vdash \varphi, \vdash \varphi \to \psi$ imply $\vdash \psi$	**T**	$K_a\varphi \to \varphi$
Sub	$\vdash \varphi \leftrightarrow \psi$ implies $\vdash \chi \leftrightarrow \chi[\varphi/\psi]$	**4**	$K_a\varphi \to K_aK_a\varphi$
		5	$\neg K_a\varphi \to K_a\neg K_a\varphi$
		Nec(K)	$\vdash \varphi$ implies $\vdash K_a\varphi$
Secrets (static)			
Own	F_aA		
PFi	$F_aB \to K_aF_aB$		
NPi	$\neg F_aB \to K_a\neg F_aB$		

Table 1: The rules and axioms of $\mathbf{G}$. In the rule **Sub** of so-called "substitution of equivalents" the expression $\chi[\varphi/\psi]$ stands for substitution of some or all occurrences of subformula φ of χ by ψ.

The basic proof system $\mathbf{G}$ is shown in Table 1. The reader may note that $\mathbf{G}$ does not depend on the call type in any way, hence the omission of the index τ. This independence of the call type is to be expected, considering that $\mathbf{G}$ is a proof system for the call-free fragment of the language.

Another observation the reader might make is that the proof system $\mathbf{G}$ is very similar to the standard proof system **S5** for modal logic. The only difference between **S5** and $\mathbf{G}$ is that the latter system includes the three extra axioms **Own**, **PFi** and **NPi** for secrets. As a result, the soundness and completeness proofs for $\mathbf{G}$ are very similar to that for **S5**. The first three lemmas are completely standard, so we omit their proofs.

Lemma 4.1 (Soundness)**.** *The proof system* $\mathbf{G}$ *is sound with respect to the class of gossip models.*

Definition 4.2. *A set* $\Gamma \subseteq \mathcal{L}^{\tau}_{[\,]\text{-free}}$ *of formulas is* consistent *if* $\Gamma \nvdash \bot$, maximal *if for every* $\varphi \in \mathcal{L}^{\tau}_{[\,]\text{-free}}$ *either* $\varphi \in \Gamma$ *or* $\neg\varphi \in \Gamma$ *and* maximal consistent *if it is both maximal and consistent.*

Definition 4.3. *Let* $\Gamma \subseteq \mathcal{L}^{\tau}_{[\,]\text{-free}}$ *be a set of formulas. Then* $K_a^{-1}\Gamma := \{\varphi \mid K_a\varphi \in \Gamma\}$.

Lemma 4.4. *If* Γ *is a maximal consistent set, then* $K_a^{-1}\Gamma$ *is consistent.*

Lemma 4.5. *If* Γ *is consistent, then there is a maximal consistent set* $\Delta \supseteq \Gamma$.

The construction of the canonical model is mostly as usual, but there is one complication: gossip models are required to be forests, where each node has exactly one $[\mathsf{c}]$-successor for every possible call c. Fortunately, this issue is reasonably easy to solve. A gossip model $M(I)$ is fully determined by its initial model I, so we can use the maximal consistent sets as the initial model, and build our canonical model on top of it. For the privacy types $\circ$ and $\boldsymbol{\ominus}$ this is reasonably straightforward, for $\bullet$ we need to do more work.

Definition 4.6. *Let* $\tau = (\mathsf{p}, \mathsf{d}, \mathsf{o})$ *with* $\mathsf{p} \in \{\circ, \boldsymbol{\ominus}\}$. *The canonical initial model* $I^{\tau} = (W^{\tau}, R^{\tau}, Q^{\tau})$ *is given by*

- W^{τ} *is the set of maximal consistent sets,*
- $R^{\tau}_a := \{(\Gamma, \Delta) \in W^{\tau} \times W^{\tau} \mid K^{-1}\Gamma \subseteq \Delta\}$,
- $Q^{\tau}(a, \Gamma) := \{B \mid F_aB \in \Gamma\}$.

The canonical model M^{τ} *is given by* $M^{\tau} := M(I^{\tau})$.

Now that we have built M^{τ} on top of I^{τ}, we can immediately forget about most of its states: every epistemic formula that is satisfied in some state of M^{τ} is already satisfied in one of its root states.

Lemma 4.7 (Truth Lemma)**.** *For every maximal consistent set* Γ *and every* $\varphi \in \mathcal{L}^{\tau}_{[\,]\text{-free}}$*, we have* $M^{\tau}, (\Gamma, \epsilon) \models \varphi$ *if and only if* $\varphi \in \Gamma$.

Proof. Because $\tau(\mathsf{p}) \in \{\circ, \boldsymbol{\ominus}\}$, a state (Γ, ϵ) can only be a-indistinguishable from another state if that state is of the form (Δ, ϵ). As a result, $M^{\tau}, (\Gamma, \epsilon) \models \varphi$ if and only if $I^{\tau}, \Gamma \models \varphi$ for every $\varphi \in \mathcal{L}^{\tau}_{[\,]\text{-free}}$.

Using the same arguments as the standard truth lemma for modal logic, we can see that $I^{\tau}, \Gamma \models \varphi$ if and only if $\varphi \in \Gamma$. Combining both equivalences, we get $M^{\tau}, (\Gamma, \epsilon) \models \varphi$ if and only if $\varphi \in \Gamma$. □

Theorem 4.8. *The proof system* $\mathbf{G}$ *is sound and strongly complete for* $\mathcal{L}^{\tau}_{[\,]\text{-free}}$ *on the class of all gossip models for call types* $(\bigcirc, \mathsf{d}, \mathsf{o})$ *and* $(\bigcirc\!\!\!\!\bullet, \mathsf{d}, \mathsf{o})$.

Now that we have a proof system for the call-free formulas, we can add extra axioms and rules for formulas with calls. These axioms can be separated into four groups. The *call basics* state some basic facts about calls. The *effect axioms* describe the secrets that agents learn through a call that they are involved in. The *observance axioms* describe the other facts that agents learn through calls that they are involved in. Finally, the *privacy axioms* describe what agents learn from calls that they were not involved in. These axioms are shown in Tables 2–5.

Call Basics	
K(c)	$[\mathsf{c}](\varphi \to \psi) \to ([\mathsf{c}]\varphi \to [\mathsf{c}]\psi)$
Fnc	$[\mathsf{c}]\neg\varphi \leftrightarrow \neg[\mathsf{c}]\varphi$
Nec(c)	$\vdash^{\tau} \varphi$ implies $\vdash^{\tau} [\mathsf{c}]\varphi$

Table 2: Basic axioms and rule for calls

Effect Axioms		
Effect for outsiders		
Ext	$[\mathsf{c}]F_aB \leftrightarrow F_aB$	$a \notin Ag(\mathsf{c})$
Effect per direction type		
Eff$_1(\triangleright)$	$[a \triangleright b]F_aD \leftrightarrow F_aD$	
Eff$_2(\triangleright)$	$[a \triangleright b]F_bD \leftrightarrow (F_aD \vee F_bD)$	
Eff$_1(\triangleleft)$	$[a \triangleleft b]F_bD \leftrightarrow F_bD$	
Eff$_2(\triangleleft)$	$[a \triangleleft b]F_aD \leftrightarrow (F_aD \vee F_bD)$	
Eff$(\diamond)$	$[a \diamond b]F_cD \leftrightarrow (F_aD \vee F_bD)$	$c \in \{a, b\}$

Table 3: Axioms for the effect of a call.

If $\tau = (\bigcirc, \mathsf{d}, \mathsf{o})$ or $\tau = (\bigcirc\!\!\!\!\bullet, \mathsf{d}, \mathsf{o})$, the proof system $\mathbf{G}^{\tau}$ is obtained by adding the relevant axioms from Tables 2–5 to the basic proof system $\mathbf{G}$. We should first convince ourselves of the soundness of this proof system.

Proposition 4.9. *The proof system* $\mathbf{G}^{\tau}$ *is sound for* $\mathcal{L}^{\tau}$ *on the class of all gossip models for call types* $\tau = (\bigcirc, \mathsf{d}, \mathsf{o})$ *and* $\tau = (\bigcirc\!\!\!\!\bullet, \mathsf{d}, \mathsf{o})$.

Proof. The axiom **K**(c) and rule **Nec**(c) are sound because $[\mathsf{c}]$ is a relation on gossip models. The axiom **Fnc** is sound because the relation $[\mathsf{c}]$ is a function. The effect axioms are a straightforward encoding of the secrets that are exchanged in the relevant call type, so we consider their soundness to be obvious.

The observance axioms represent a kind of hypothetical reasoning. We look at the axiom **Obs**$^{\tau}(\triangleright, \alpha)$ in detail. With observation level α, the information that the recipient

Axioms for observance level β for sender ($\mathbf{Obs}^s$) and receiver ($\mathbf{Obs}^r$)		
$\mathbf{Obs}^s(\triangleright, \beta)$	$[a \triangleright b]K_a\varphi \leftrightarrow K_a[a \triangleright b]\varphi$	
$\mathbf{Obs}^r(\triangleright, \beta)$	$[a \triangleright b]K_b\varphi \leftrightarrow \bigvee_{\mathsf{Q}\subseteq S}(O_a\mathsf{Q} \wedge K_b(O_a\mathsf{Q} \to [a \triangleright b]\varphi))$	
$\mathbf{Obs}^s(\triangleleft, \beta)$	$[a \triangleleft b]K_b\varphi \leftrightarrow K_b[a \triangleleft b]\varphi$	
$\mathbf{Obs}^r(\triangleleft, \beta)$	$[a \triangleleft b]K_a\varphi \leftrightarrow \bigvee_{\mathsf{Q}\subseteq S}(O_b\mathsf{Q} \wedge K_a(O_b\mathsf{Q} \to [a \triangleleft b]\varphi))$	
$\mathbf{Obs}^{s,r}(\diamond, \beta)$	$[a \diamond b]K_c\varphi \leftrightarrow \bigvee_{\mathsf{Q},\mathsf{R}\subseteq S}(O_a\mathsf{Q} \wedge O_b\mathsf{R} \wedge K_c((O_a\mathsf{Q} \wedge O_b\mathsf{R}) \to [a \diamond b]\varphi))$	$c \in \{a, b\}$
Axioms for observance level α for sender ($\mathbf{Obs}^s$) and receiver ($\mathbf{Obs}^r$)		
$\mathbf{Obs}^s(\triangleright, \alpha)$	$[a \triangleright b]K_a\varphi \leftrightarrow K_a[a \triangleright b]\varphi$	
$\mathbf{Obs}^r(\triangleright, \alpha)$	$[a \triangleright b]K_b\varphi \leftrightarrow \bigvee_{\mathsf{Q}\subseteq S}(O_{ab}\mathsf{Q} \wedge K_b(O_{ab}\mathsf{Q} \to [a \triangleright b]\varphi))$	
$\mathbf{Obs}^s(\triangleleft, \alpha)$	$[a \triangleleft b]K_b\varphi \leftrightarrow K_b[a \triangleleft b]\varphi$	
$\mathbf{Obs}^r(\triangleleft, \alpha)$	$[a \triangleleft b]K_a\varphi \leftrightarrow \bigvee_{\mathsf{Q}\subseteq S}(O_{ab}\mathsf{Q} \wedge K_a(O_{ab}\mathsf{Q} \to [a \triangleleft b]\varphi))$	
$\mathbf{Obs}^{s,r}(\diamond, \alpha)$	$[a \diamond b]K_c\varphi \leftrightarrow \bigvee_{\mathsf{Q}\subseteq S}(O_{ab}\mathsf{Q} \wedge K_c(O_{ab}\mathsf{Q} \to [a \diamond b]\varphi))$	$c \in \{a, b\}$

Table 4: Axioms about observance.

Privacy Axioms		
$\mathbf{Pri}(\bigcirc)$	$[\mathsf{c}]K_a\varphi \leftrightarrow K_a[\mathsf{c}]\varphi$	$a \notin Ag(\mathsf{c})$
$\mathbf{Pri}(\text{◓})$	$[\mathsf{c}]K_a\varphi \leftrightarrow K_a \bigwedge_{\{\mathsf{c}' \in \mathsf{C}^{\mathsf{d}} \mid a \notin Ag(\mathsf{c}')\}} [\mathsf{c}']\varphi$	$a \notin Ag(\mathsf{c})$

Table 5: Axioms for the the effect of privacy.

b of a call $a \triangleright b$ learns is the set of secrets known to either agent before the call, i.e., the set Q such that $O_{ab}\mathsf{Q}$. As such, we have $[a \triangleright b]K_b\varphi$ if and only if before this call, b could already predict that if the shared set of secrets was Q, then φ would be true after the call. Conditioning on the set of shared secrets can be done using a disjunction over all possibilities, so we have $[a \triangleright b]K_b\varphi \leftrightarrow \bigvee_{\mathsf{Q}\subseteq S}(O_{ab}\mathsf{Q} \wedge K_b(O_{ab}\mathsf{Q} \to [a \triangleright b]\varphi))$. So the axiom is sound.

More formally, recall that with this call type we have $(w, \vec{\mathbf{c}}.a \triangleright b) \sim_b (w', \vec{\mathbf{c}}'.\mathsf{c})$ if and only if $\mathsf{c} = a \triangleright b$, $(w, \vec{\mathbf{c}}) \sim_b (w', \vec{\mathbf{c}}')$ and the set of secrets known to either a or b is the same in $(w, \vec{\mathbf{c}})$ and $(w', \vec{\mathbf{c}}')$. As a result, we have the following chain of

equivalences.

$$
\begin{aligned}
& M,(w,\vec{\mathsf{c}}) \models [a \triangleright b] K_b \varphi \\
\Leftrightarrow\ & M,(w,\vec{\mathsf{c}}.a \triangleright b) \models K_b \varphi \\
\Leftrightarrow\ & \forall (w',\vec{\mathsf{c}}'.\mathsf{c}) \sim_b (w,\vec{\mathsf{c}}.a \triangleright b) : M,(w',\vec{\mathsf{c}}'.\mathsf{c}) \models \varphi \\
\Leftrightarrow\ & \forall (w',\vec{\mathsf{c}}') \sim_b (w,\vec{\mathsf{c}}) : \text{ if } \mathsf{Q}_{ab}(w,\vec{\mathsf{c}}) = \mathsf{Q}_{ab}(w',\vec{\mathsf{c}}') \text{ then } M,(w',\vec{\mathsf{c}}'.a \triangleright b) \models \varphi \\
\Leftrightarrow\ & \forall (w',\vec{\mathsf{c}}') \sim_b (w,\vec{\mathsf{c}}) : M,(w',\vec{\mathsf{c}}') \models O_{ab}\mathsf{Q}(w,\vec{\mathsf{c}}) \to [a \triangleright b]\varphi \\
\Leftrightarrow\ & M,(w,\vec{\mathsf{c}}) \models K_b(O_{ab}\mathsf{Q}(w,\vec{\mathsf{c}}) \to [a \triangleright b]\varphi) \\
\Leftrightarrow\ & M,(w,\vec{\mathsf{c}}) \models \bigvee_{\mathsf{Q} \subseteq S} (O_{ab}\mathsf{Q} \wedge K_b(O_{ab}\mathsf{Q} \to [a \triangleright b]\varphi))
\end{aligned}
$$

So $\mathbf{Obs}^r(\triangleright, \alpha)$ is sound. Soundness of the other observance axioms can be shown similarly.

Finally, consider the privacy axioms. Like the observance axioms, these represent hypothetical reasoning. With privacy type $\bigcirc$, if a call c with $a \notin Ag(\mathsf{c})$ takes place, then a learns that this call took place, but no more. This means that a will know φ after the call if and only if before the call a knew that if the call were to take place, then φ would become true. So with privacy $\bigcirc$ we have $[\mathsf{c}]K_a\varphi \leftrightarrow K_a[\mathsf{c}]\varphi$.

With privacy type $\blacksquare$, agent a does not learn that the call c took place. Instead, a only learns that some call took place that they were not involved in. This means that a will know φ after the call if and only if before the call a knew that any call not involving them would result in φ becoming true. So with privacy $\blacksquare$ we have $[\mathsf{c}]K_a\varphi \leftrightarrow K_a \bigwedge_{\{\mathsf{c}' \in \mathsf{C}^\mathsf{d} \mid a \notin Ag(\mathsf{c}')\}} [\mathsf{c}']\varphi$. □

One important property of these axioms is that they are *reduction axioms* for the operator $[\mathsf{c}]$. That is to say, these axioms are equivalences[4] with the property that the formulas inside the scope of a $[\mathsf{c}]$ operator on the right hand side (if any) are strict subformulas of those on the left hand side. This yields the following theorem.

Theorem 4.10. *Let $\tau = (\bigcirc, \mathsf{d}, \mathsf{o})$ or $\tau = (\blacksquare, \mathsf{d}, \mathsf{o})$. Then for every formula $\varphi \in \mathcal{L}\tau$ there is a formula $\psi \in \mathcal{L}^\tau_{[\,]\text{-free}}$ such that $\vdash^\tau \varphi \leftrightarrow \psi$.*

Proof. We give a detailed proof for the case $\tau = (\bigcirc, \triangleright, \alpha)$. First, note that because $\{\neg, \to\}$ is truth-functionally complete, we can assume without loss of generality that φ contains only these Boolean connectives. Secondly, because **Sub** allows us to substitute provably equivalent formulas, it suffices to show the theorem for formulas of the form $\varphi = [a \triangleright b]\chi$, where $\chi \in \mathcal{L}^\tau_{[\,]\text{-free}}$.

We now prove the theorem by induction on the construction of χ. As base case, suppose that χ is an atom, so $\varphi = [a \triangleright b]F_cD$. If $c \neq b$, then by **Ext** or $\mathbf{Eff}_1(\triangleright)$ we have $\vdash^\tau [a \triangleright b]F_cD \leftrightarrow F_cD$. If $c = b$, then by $\mathbf{Eff}_2(\triangleright)$ we have $\vdash^\tau [a \triangleright b]F_cD \leftrightarrow (F_aD \vee F_cD)$. In either case, φ is provably equivalent to a formula $\psi \in \mathcal{L}^\tau_{[\,]\text{-free}}$.

Suppose then as induction hypothesis that χ is not atomic, and that for every strict subformula χ' of χ and for every c, d the formula $[c \triangleright d]\chi'$ is provably equivalent to some $\psi' \in \mathcal{L}^\tau_{[\,]\text{-free}}$. We continue with a case distinction on the main connective of χ.

[4] Except for $\mathbf{K}(\mathsf{c})$, but $[\mathsf{c}](\varphi \to \psi) \leftrightarrow ([\mathsf{c}]\varphi \to [\mathsf{c}]\psi)$ is provable using $\mathbf{K}(\mathsf{c})$ and **Fnc**.

- Suppose $\varphi = [a \triangleright b]\neg\chi'$. Axiom **Fnc** yields $\vdash^\tau [a \triangleright b]\neg\chi' \leftrightarrow \neg[a \triangleright b]\chi'$. By the induction hypothesis $\vdash^\tau [a \triangleright b]\chi' \leftrightarrow \psi'$ so, using **Sub**, we have $\vdash^\tau [a \triangleright b]\neg\chi' \leftrightarrow \neg\psi'$.

- Suppose $\varphi = [a \triangleright b](\chi' \to \chi'')$. Using $\mathbf{K}(\mathsf{c})$ and **Fnc**, we have $\vdash^\tau [a \triangleright b](\chi' \to \chi'') \leftrightarrow ([a \triangleright b]\chi' \to [a \triangleright b]\chi'')$. By the induction hypothesis, $\vdash^\tau [a \triangleright b]\chi' \leftrightarrow \psi'$ and $\vdash^\tau [a \triangleright b]\chi'' \leftrightarrow \psi''$. Using **Sub**, this yields $\vdash^\tau [a \triangleright b](\chi' \to \chi'') \leftrightarrow (\psi' \to \psi'')$.

- Suppose $\varphi = [a \triangleright b]K_c\chi'$, where $c \neq b$. Then using either **Pri**($\bigcirc$) or $\mathbf{Obs}^s(\triangleright, \alpha)$ we have $\vdash^\tau [a \triangleright b]K_c\chi' \leftrightarrow K_c[a \triangleright b]\chi'$. By the induction hypothesis, $\vdash^\tau [a \triangleright b]\chi' \leftrightarrow \psi'$ and therefore, using **Sub**, $\vdash^\tau [a \triangleright b]K_c\chi' \leftrightarrow K_c\psi'$.

- Suppose $\varphi = [a \triangleright b]K_b\chi'$. Then $\mathbf{Obs}^r(\triangleright, \alpha)$ states that $\vdash^\tau [a \triangleright b]K_b\chi' \leftrightarrow \bigvee_{\mathsf{Q} \subseteq S}(O_{ab}\mathsf{Q} \wedge K_b(O_{ab}\mathsf{Q} \to [a \triangleright b]\chi'))$. By the induction hypothesis, $\vdash^\tau [a \triangleright b]\chi' \leftrightarrow \psi'$. So using **Sub**, we have $\vdash^\tau [a \triangleright b]K_b\chi' \leftrightarrow \bigvee_{\mathsf{Q} \subseteq S}(O_{ab}\mathsf{Q} \wedge K_b(O_{ab}\mathsf{Q} \to \psi'))$.

These are all possible main connectives of χ, so this completes the induction. The proofs for the other call types are very similar, using the appropriate axioms. So we omit those cases here.

□

Completeness of $\mathbf{G}^\tau$ for $\mathcal{L}^\tau$ now follows immediately.

Corollary 4.11. *The proof system $\mathbf{G}^\tau$ is strongly complete for $\mathcal{L}^\tau$ on the class of all gossip models for call types $\tau = (\bigcirc, \mathsf{d}, \mathsf{o})$ and $\tau = (\text{◓}, \mathsf{d}, \mathsf{o})$.*

Proof. Let $\Gamma \subseteq \mathcal{L}^\tau$ and $\varphi \in \mathcal{L}^\tau$ be such that $\Gamma \models^\tau \varphi$. Let $\varphi' \in \mathcal{L}^\tau_{[\,]\text{-free}}$ and $\Gamma' \subseteq \mathcal{L}^\tau_{[\,]\text{-free}}$ be such that $\vdash^\tau \varphi \leftrightarrow \varphi'$ and $\Gamma' = \{\gamma' \mid \exists \gamma \in \Gamma : \vdash^\tau \gamma \leftrightarrow \gamma'\}$. Then $\Gamma' \models \varphi'$ and therefore, by completeness of $\mathbf{G}$ for $\mathcal{L}^\tau_{[\,]\text{-free}}$ (Theorem 4.8), $\Gamma' \vdash \varphi'$. Since $\mathbf{G}$ is a fragment of $\mathbf{G}^\tau$, this implies $\Gamma' \vdash^\tau \varphi'$. Furthermore, using the provable equivalence of φ' and φ as well as γ' and γ, we get $\Gamma \vdash^\tau \varphi$. □

We have now shown soundness and completeness.

Theorem 4.12. *The proof system $\mathbf{G}^\tau$ is sound and strongly complete for $\mathcal{L}^\tau$ on the class of all gossip models for call types $\tau = (\bigcirc, \mathsf{d}, \mathsf{o})$ and $\tau = (\text{◓}, \mathsf{d}, \mathsf{o})$.*

5 Axiomatisation: Asynchronous Case

In Section 4, we introduced axiomatizations for synchronous call types, with privacy types $\bigcirc$ or ◓. Here, we consider the asynchronous privacy type ●. We therefore assume throughout the section that the privacy type is ●. The first thing to note is that the observance axioms of Table 4 are *not* sound in the asynchronous case. Take for example the formula $[a \triangleright b]K_a\neg F_cA \leftrightarrow K_a[a \triangleright b]\neg F_cA$, which is an instance of axiom $\mathbf{Obs}^s(\triangleright, \alpha)$. But now consider the rooted model M, ϵ. We have $M, \epsilon \models K_a[a \triangleright b]\neg F_cA$, because a knows that the single call $a \triangleright b$ will not, by itself, teach c the secret A. But after the call $a \triangleright b$ has happened, agent a cannot be certain that this is the only call that happened; we

have $a \triangleright b \sim_a a \triangleright b.b \triangleright c$. Because $M, a \triangleright b.b \triangleright c \not\models \neg F_c A$, we have $M, a \triangleright b \not\models K_a \neg F_c A$ and therefore $M, \epsilon \not\models [a \triangleright b] K_a \neg F_c A$. So the axiom $\mathbf{Obs}^s(\triangleright, \alpha)$ is not satisfied.

The problem lies in the fact that in the asynchronous case agent a always considers it possible that any number of calls (not involving a) happened after the call $a \triangleright b$. So while

$$[a \triangleright b] K_a \varphi \leftrightarrow K_a [a \triangleright b] \varphi$$

is unsound,

$$[a \triangleright b] K_a \varphi \leftrightarrow K_a \bigwedge_{\{\vec{\mathbf{c}} \mid a \notin Ag(\vec{\mathbf{c}})\}} [a \triangleright b.\vec{\mathbf{c}}] \varphi$$

is sound. Sound variants of the other observance axioms can be obtained similarly. Unfortunately, $\{\vec{\mathbf{c}} \mid a \notin Ag(\vec{\mathbf{c}})\}$ is an infinite set, so this sound version of the axiom is not technically a formula. However, it turns out that we do not need to consider all sequences not containing a, a finite subset of them will suffice.

We first use an n-bisimilarity argument to give a bound that works for every call direction $(\triangleright, \triangleleft, \diamond)$ and both observance types (β, α). This bound is rather large, however, so we also give an improved bound that applies to most, but not all, combinations of direction and observance.

5.1 A general bound using n-bisimilarity

For any direction and observance type, we can now leverage preservation of n-bisimilarity (see Theorem 3.15) to limit the number of call sequences that we need to consider. For example, we have the following.

Proposition 5.1. *Let $\tau = (\bullet, \triangleright, \alpha)$. Furthermore, let $\varphi \in \mathcal{L}^{\tau}_{[\,]\text{-free}}$ be a formula of depth n. Then*

$$\models^{\tau} [a \triangleright b] K_a \varphi \leftrightarrow K_a \bigwedge_{\vec{\mathbf{c}} \in \mathcal{C}} [a \triangleright b.\vec{\mathbf{c}}] \varphi,$$

where $\mathcal{C} = \{\vec{\mathbf{c}} \mid a \notin Ag(\vec{\mathbf{c}})$ and $\ell(\vec{\mathbf{c}}) \leq f(n)\}$ and $f(n)$ is the number of equivalence classes with respect to $\leftrightsquigarrow_n$.

Proof. We start from left to right, so suppose that $M, (w, \vec{\mathbf{d}}) \models [a \triangleright b] K_a \varphi$. To show is that $M, (w, \vec{\mathbf{d}}) \models K_a \bigwedge_{\vec{\mathbf{c}} \in \mathcal{C}} [a \triangleright b.\vec{\mathbf{c}}] \varphi$. So take any $(w', \vec{\mathbf{e}}) \sim_a (w, \vec{\mathbf{d}})$. Then for any $\vec{\mathbf{c}}$ such that $a \notin Ag(\vec{\mathbf{c}})$ we also have $(w', \vec{\mathbf{e}}.a \triangleright b.\vec{\mathbf{c}}) \sim_a (w, \vec{\mathbf{d}}.a \triangleright b)$, since neither the $a \triangleright b$ call nor any of the calls not involving a allow the agent to distinguish between the two sequences. Since $M, (w, \vec{\mathbf{d}}) \models [a \triangleright b] K_a \varphi$, it follows that $M, (w', \vec{\mathbf{d}}.a \triangleright b.\vec{\mathbf{c}}) \models \varphi$. Since this holds for every $\vec{\mathbf{c}}$ with $a \notin Ag(\vec{\mathbf{c}})$, we have $M, (w, \vec{\mathbf{e}}) \models \bigwedge_{\vec{\mathbf{c}} \in \mathcal{C}} [a \triangleright b.\vec{\mathbf{c}}] \varphi$, which was to be shown.

Left to show is right to left, so suppose that $M, (w, \vec{\mathbf{d}}) \models K_a \bigwedge_{\vec{\mathbf{c}} \in \mathcal{C}} [a \triangleright b.\vec{\mathbf{c}}] \varphi$. To show is that $M, (w, \vec{\mathbf{d}}) \models [a \triangleright b] K_a \varphi$. So take any $(w', \vec{\mathbf{e}}) \sim_a (w, \vec{\mathbf{d}}.a \triangleright b)$. Then $\vec{\mathbf{e}}$ must be of the form $(\vec{\mathbf{e}}_1.a \triangleright b.\vec{\mathbf{e}}_2)$ such that $(w, \vec{\mathbf{d}}) \sim_a (w', \vec{\mathbf{e}}_1)$ and $a \notin Ag(\vec{\mathbf{e}}_2)$. We make a case distinction on $\ell(\vec{\mathbf{e}}_2)$.

If $\ell(\vec{\mathbf{e}}_2) \leq f(n)$, then $\vec{\mathbf{e}}_2 \in \mathcal{C}$. It therefore follows from $M, (w, \vec{\mathbf{d}}) \models K_a \bigwedge_{\vec{\mathbf{c}} \in \mathcal{C}} [a \triangleright b.\vec{\mathbf{c}}] \varphi$ that $M, (w', \vec{\mathbf{e}}_1.a \triangleright b.\vec{\mathbf{e}}_2) \models \varphi$.

If $\ell(\vec{\mathbf{e}}_2) > f(n)$, then by the pigeonhole principle there must be at least two calls c_1 and c_2 such that $\vec{\mathbf{e}}_2 = \vec{\mathbf{f}}_1.\mathsf{c}_1.\vec{\mathbf{f}}_2.\mathsf{c}_2.\vec{\mathbf{f}}_3$ and $\vec{\mathbf{e}}_1.a \triangleright b.\vec{\mathbf{f}}_1.\mathsf{c}_1 \leftrightsquigarrow_n \vec{\mathbf{e}}_1.a \triangleright b.\vec{\mathbf{f}}_1.\mathsf{c}_1.\vec{\mathbf{f}}_2.\mathsf{c}_2$. By Theorem 3.15 this implies that $\vec{\mathbf{e}}_1.a \triangleright b.\vec{\mathbf{f}}_1.\mathsf{c}_1.\vec{\mathbf{f}}_3 \leftrightsquigarrow_n \vec{\mathbf{e}}$. As long as $\ell(\vec{\mathbf{f}}_1.\mathsf{c}_1.\vec{\mathbf{f}}_3) > f(n)$ we can repeat this procedure, until we arrive at a sequence $\vec{\mathbf{c}}$ of length at most $f(n)$ such that $\vec{\mathbf{e}}_1.a \triangleright b.\vec{\mathbf{c}} \leftrightsquigarrow_b \vec{\mathbf{e}}$.

Since $\vec{\mathbf{c}}$ is obtained from $\vec{\mathbf{e}}$ by removing calls, it cannot contain any calls involving a. We therefore have $\vec{\mathbf{c}} \in \mathcal{C}$. So it follows from $M, (w, \vec{\mathbf{d}}) \models K_a \bigwedge_{\vec{\mathbf{c}} \in \mathcal{C}} [a \triangleright b.\vec{\mathbf{c}}]\varphi$ that $M, (w', \vec{\mathbf{e}}_1.a \triangleright b.\vec{\mathbf{c}}) \models \varphi$. Then, because φ is a modal formula of depth n and the two states are n-bismilar, we obtain $M, (w', \vec{\mathbf{e}}) \models \varphi$.

In either case of our case distinction, we arrived at $M, (w', \vec{\mathbf{e}}) \models \varphi$. Since this holds for every $(w', \vec{\mathbf{e}}) \sim_a (w, \vec{\mathbf{d}}.a \triangleright b)$, we get $M, (w, \vec{\mathbf{d}}) \models [a \triangleright b] K_a \varphi$. □

Soundness of the axioms for other call types can be shown similarly. Unfortunately, while this yields a sound and complete axiomatization, the axioms in question are rather large: we need to consider call sequences of length up to $f(n)$, which is a non-elementary bound. Specifically, $f(0) = 2^{|P|}$ and $f(n+1) = 2^{|P|} \cdot 2^{f(n)}$. It therefore seems useful to look for a smaller bound.

5.2 A formula-independent bound for most call types

An agent may gain information about another agent's knowledge of secrets either directly or indirectly. Direct knowledge is obtained by either telling another agent a secret, or being told a secret by another agent. If a tells b secret C, then a knows that b knows C, and likewise b knows that a knows C. Indirect knowledge is obtained by reasoning about the knowledge of an agent not directly involved in a call. For example, if the call direction is $\diamond$, and a tells b exactly the secrets A and C, then b can conclude that c knows the secret A.

In order to create our bound, we will look at calls that do not increase the direct knowledge of either participant.

Definition 5.2. *Let a state $(w, \vec{\mathbf{c}})$ be given. Agent a has* direct knowledge *of agent b's knowledge of secret C in this state, denoted $DK_a(F_bC)$, if one of the following three conditions holds:*

1. *the sequence $\vec{\mathbf{c}}$ contains a call ab or ba where a is a sender, and where a knew secret C before the call,*
2. *the sequence $\vec{\mathbf{c}}$ contains a call ab or ba where a is a receiver, b knew secret C before the call and the observance level is β,*
3. *the sequence $\vec{\mathbf{c}}$ contains a call ab or ba where a is a receiver, b knew secret C before the call and a did not know C before the call.*

We abuse notation by treating $DK_a(F_bC)$ as a formula.

A call ab increases direct knowledge for agent a *at a state $(w, \vec{\mathbf{c}})$ if there is some C such that $(w, \vec{\mathbf{c}}) \models \neg DK_a(F_bC) \wedge [ab] DK_a(F_bC)$. A call ab* increases direct knowledge *if it increases direct knowledge for either agent.*

When the state is clear from context, we just say that a call increases direct knowledge. The following lemma states a few properties of direct knowledge that will be important.

Lemma 5.3.

1. *If a call does not increase direct knowledge, then no new secrets are learned in the call.*
2. *Agents know what their direct knowledge is (i.e., if* $(w_1, \vec{\mathbf{c}}_1) \sim_a (w_2, \vec{\mathbf{c}}_2)$ *then for every* b *and* C, $w_1, \vec{\mathbf{c}}_1 \models DK_a(F_bC) \Leftrightarrow w_2, \vec{\mathbf{c}}_2 \models DK_a(F_bC)$*).*
3. *If the observance level is* β*, then a call* ab *increases the direct knowledge of* a *if and only if it increases the direct knowledge of* b*.*
4. *A call* $a \diamond b$ *increases the direct knowledge of* a *if and only if it increases the direct knowledge of* b*.*

Proof.

1. Trivial.
2. A difference in direct knowledge for a between $(w_1, \vec{\mathbf{c}}_1)$ and $(w_2, \vec{\mathbf{c}}_2)$ would require there to be a call ab in $\vec{\mathbf{c}}_1$ and $\vec{\mathbf{c}}_2$ with the property that (i) the set of secrets sent by a to b differs between the states or (ii) a observes a difference in the set of secrets received from b between the two states. Either of (i) and (ii) would be sufficient to conclude $(w_1, \vec{\mathbf{c}}_1) \not\sim_a (w_2, \vec{\mathbf{c}}_2)$, so the direct knowledge of a must be the same in the two states.
3. Suppose that the call ab increases the direct knowledge of a. Then either a sends b a secret C that was not communicated between the two agents before, or a observes b's sending of secret C. In the first case, b observes a's sending of C, in the second case b sends a secret C. In either case, b also gains direct knowledge of a knowing C.
4. Suppose that $a \diamond b$ increases the direct knowledge for either agent. Then there is some secret C that is being communicated between the two agents for the first time in this call. So at least one of the agents must have learned C since the last time (if any) that a and b communicated. If only one of them learned C before the $a \diamond b$ call, then that agent gains new direct knowledge by sending the new secret C and the other agent gains new direct knowledge by observing that C is being sent to them. If both agents learned C before the $\diamond$ call, then they both gain direct knowledge by sending the new secret C.

□

Corollary 5.4. *After any call* ab*, if the observance is* β *or the direction is* $\diamond$*, both agents know whether the call created new direct knowledge.*

Proof. The β case follows from points 2 and 3 of Lemma 5.3, the $\diamond$ case follows from points 2 and 4 of the same lemma. □

Using this corollary, we can find our improved bound.

Proposition 5.5. *Let $\tau = (\bullet, \mathsf{d}, \beta)$ or $\tau = (\bullet, \diamond, \alpha)$. Furthermore, let $\vec{\mathbf{c}} = \vec{\mathbf{c}}_1.ab.\vec{\mathbf{c}}_2.ab.\vec{\mathbf{c}}_3.ab.\vec{\mathbf{c}}_4$ and $\vec{\mathbf{d}} = \vec{\mathbf{c}}_1.ab.\vec{\mathbf{c}}_2\vec{\mathbf{c}}_3.ab.\vec{\mathbf{c}}_4$. If $(w, \vec{\mathbf{c}})$ is such that a and b learn no direct knowledge about each other during the $\vec{\mathbf{c}}_2.ab.\vec{\mathbf{c}}_3.ab$ segment, then for every $\varphi \in \mathcal{L}^{\tau}_{[\,]\text{-free}}$, we have $w, \vec{\mathbf{c}} \models \varphi$ if and only if $w, \vec{\mathbf{d}} \models \varphi$.*

Proof. By induction on the construction of φ. The extra ab call in $\vec{\mathbf{c}}$ did not increase direct knowledge, so in particular it did not change the set of secrets known to any agent. So an atomic formula cannot distinguish between $(w, \vec{\mathbf{c}})$ and $(w, \vec{\mathbf{d}})$. Suppose then as induction hypothesis that φ is not atomic and that the proposition holds for all strict subformulas of φ.

A Boolean combination of formulas cannot distinguish between two states unless at least one of the combined formulas does. We can therefore assume without loss of generality that φ if of the form $\hat{K}_c\psi$ for some agent c.

Suppose $c \notin \{a, b\}$. The extra ab call does not change the secrets known to any agent, so for agents not directly involved in the call it is impossible to ever know whether the call happened. We therefore have $(w, \vec{\mathbf{c}}) \sim_c (w, \vec{\mathbf{d}})$, from which it follows immediately that $\hat{K}_c\psi$ does not distinguish between the two states.

Suppose then that $c \in \{a, b\}$.

- Suppose that $w, \vec{\mathbf{c}} \models \hat{K}_c\psi$. Then there is a state $(w', \vec{\mathbf{c}}') \sim_c (w, \vec{\mathbf{c}})$ such that $w', \vec{\mathbf{c}}' \models \psi$. Because the two states are c-indistinguishable, they must have the same c-reduction. So $\vec{\mathbf{c}}' = \vec{\mathbf{c}}'_1.ab.\vec{\mathbf{c}}'_2.ab.\vec{\mathbf{c}}'_3.ab.\vec{\mathbf{c}}'_4$. Take $\vec{\mathbf{d}}' = \vec{\mathbf{c}}'_1.ab.\vec{\mathbf{c}}'_2.ab.\vec{\mathbf{c}}'_3.ab.\vec{\mathbf{c}}'_4$.

  Agents know when they gained direct knowledge, so c did not gain direct knowledge about the other agent in the $\vec{\mathbf{c}}'_2.ab.\vec{\mathbf{c}}'_3.ab$ segment. If a call increases direct knowledge for one agent it does so for the other agent as well, so the other agent does not gain direct knowledge about c during $\vec{\mathbf{c}}'_2.ab.\vec{\mathbf{c}}'_3.ab$ either.

  The conditions of the proposition are therefore satisfied for $(w', \vec{\mathbf{c}}')$ and $(w', \vec{\mathbf{d}}')$. By the induction hypothesis, this implies that $w', \vec{\mathbf{d}}' \models \psi$. Note also that the c-reductions of $\vec{\mathbf{d}}'$ and $\vec{\mathbf{d}}$ are the same. So in order for c to distinguish between $(w, \vec{\mathbf{d}})$ and $(w', \vec{\mathbf{d}}')$, there would need to be at least one call c in the c-reduction where c observes different secrets in $(w, \vec{\mathbf{d}})$ and $(w', \vec{\mathbf{d}}')$. But that is impossible: the secrets observed by c in c (i) are the same in $(w, \vec{\mathbf{d}})$ and $(w, \vec{\mathbf{c}})$, because the extra ab call did not change the set of secrets known to any agent, (ii) are the same in $(w, \vec{\mathbf{c}})$ and $(w', \vec{\mathbf{c}}')$, because $(w, \vec{\mathbf{c}}) \sim_c (w', \vec{\mathbf{c}}')$ and (iii) are the same in $(w', \vec{\mathbf{c}}')$ and $(w', \vec{\mathbf{d}}')$, because the extra ab call did not change the set of secrets known to any agent. We must therefore have $(w', \vec{\mathbf{d}}') \sim_c (w, \vec{\mathbf{d}})$, and therefore $w, \vec{\mathbf{d}} \models \hat{K}_c\psi$.

- Suppose that $w, \vec{\mathbf{d}} \models \hat{K}_c\psi$. Then there is a state $(w', \vec{\mathbf{d}}') \sim_c (w, \vec{\mathbf{d}})$ such that $w', \vec{\mathbf{d}}' \models \psi$. Because the two states are c-indistinguishable, they must have the same c-reduction. So there are $\vec{\mathbf{c}}'_1$, $\vec{\mathbf{c}}'_2, \vec{\mathbf{c}}'_3$ and $\vec{\mathbf{c}}'_4$ such that $\vec{\mathbf{d}}' =$

$\vec{\mathbf{c}}'_1.ab.\vec{\mathbf{c}}'_2.\vec{\mathbf{c}}'_3.ab.\vec{\mathbf{c}}'_4$, where for every $i \in \{1,2,3,4\}$, the c-reductions of $\vec{\mathbf{c}}_i$ and $\vec{\mathbf{c}}'_1$ are the same.[5] Take $\vec{\mathbf{c}}' = \vec{\mathbf{c}}'_1.ab.\vec{\mathbf{c}}'_2.ab.\vec{\mathbf{c}}'_3.ab.\vec{\mathbf{c}}'_4$.

From the fact that neither a nor b gained direct knowledge about the other agent in the $\vec{\mathbf{c}}_2.ab.\vec{\mathbf{c}}_3.ab$ segment of $\vec{\mathbf{c}}$, it follows that they did not gain direct knowledge about each other in the $\vec{\mathbf{c}}_2.\vec{\mathbf{c}}_3.ab$ either. Furthermore, agents know when they gained direct knowledge, so c did not gain direct knowledge about the other agent during $\vec{\mathbf{c}}'_2.\vec{\mathbf{c}}'_3.ab$. If a call increases direct knowledge for one agent it does so for the other agent as well, so the other agent does not gain direct knowledge about c during $\vec{\mathbf{c}}'_2.\vec{\mathbf{c}}'_3.ab$ either.

From the fact that the second call ab in $\vec{\mathbf{d}}'$ did not increase direct knowledge, it follows that the sender(s) in that call did not gain any new secrets since the previous ab call. As a result, an extra ab call inserted into the sequence would not increase direct knowledge for either agent, nor would it change the set of secrets known to anyone. So the conditions of the proposition are satisfied for $\vec{\mathbf{c}}'$ and $\vec{\mathbf{d}}'$. By the induction hypothesis, it follows that $w', \vec{\mathbf{c}}' \models \psi$.

Note also that we chose $\vec{\mathbf{c}}'_2$ and $\vec{\mathbf{c}}'_3$ in such a way that the c-reduction of $\vec{\mathbf{c}}'$ is the same as that of $\vec{\mathbf{c}}$. Furthermore, by similar reasoning as in the previous case, it follows that at every call in the c-reduction, the same secrets are observed by c in the two sequences. So $(w, \vec{\mathbf{c}}) \sim_c (w', \vec{\mathbf{c}}')$. We therefore have $w, \vec{\mathbf{c}} \models \hat{K}_c\psi$.

The above two cases show that $\hat{K}_c\psi$ does not distinguish between the two states. This concludes the induction step and thereby the proof. □

Corollary 5.6. *If $\tau = (\bullet, \mathsf{d}, \beta)$ or $\tau = (\bullet, \diamond, \alpha)$, then for every call sequence $\vec{\mathbf{c}}$ there is a call sequence $\vec{\mathbf{c}}'$ such that (i) for every w, $(w, \vec{\mathbf{c}})$ and $(w, \vec{\mathbf{c}}')$ satisfy the same formulas and (ii) $\ell(\vec{\mathbf{c}}') \leq 2|Ag|^3$.*

Proof. Let $\vec{\mathbf{c}}'$ be the shortest call sequence that satisfies the same formulas as $\vec{\mathbf{c}}$. Take any two agents a and b, and consider the number of calls ab in $\vec{\mathbf{c}}'$. By Proposition 5.5 and the minimality of $\vec{\mathbf{c}}'$, the direct knowledge of a and b about each other must increase at least every second ab call, since otherwise the proposition would allow us to remove one call. Note furthermore that any call that increases direct knowledge must do so for both agents. There are $|Ag|$ different pieces of direct knowledge, one for each secret C, that a can have about b, another $|Ag|$ pieces that b can have about a. So there can be at most $|Ag|$ calls ab where direct knowledge increases. There are $|Ag|^2$ different pairs of agents, so the total number of calls in $\vec{\mathbf{c}}'$ is at most $2|Ag|^3$. □

The general bound shown in the previous subsection shows that for any formula φ, if φ true after any number of calls, then it is true after a number of calls that is exponential in $|Ag|$ and non-elementary in the depth of φ. Our improved bound for $(\bullet, \mathsf{d}, \beta)$ and $(\bullet, \diamond, \alpha)$ is polynomial in $|Ag|$ and independent of φ: $2|Ag|^3$ calls is enough to for any formula, regardless of the formula's depth.

[5] Note that $\vec{\mathbf{c}}'_2$ and $\vec{\mathbf{c}}'_3$ need not be the unique sequences with these properties. For example, if $\vec{\mathbf{c}}'_2 = \vec{\mathbf{e}}$ and $\vec{\mathbf{c}}'_3 = \mathsf{c}.\vec{\mathbf{f}}$, where $c \notin Ag(\mathsf{c})$, then $\vec{\mathbf{e}}.\mathsf{c}$ and $\vec{\mathbf{f}}$ also have the required properties.

5.3 No formula-independent bound for the remaining call types

For the remaining two call types, $(\bullet, \triangleright, \alpha)$ and $(\bullet, \triangleleft, \alpha)$, we cannot find a bound that does not depend on φ. The following lemma shows that we can construct infinitely many non-equivalent formulas. This suffices to show that there can be no formula-independent bound on the length of call sequences that we need to consider.

Lemma 5.7. *Let $m \leq n$ and let $\vec{\mathbf{c}}_n^m$ be a call sequence containing n alternations $a \triangleright b, b \triangleright a$ with a single $c \triangleright a$ call after m such iterations. Furthermore, let φ_k be given by $\varphi_0 = F_{ab}C$ and $\varphi_{i+1} = K_a K_b \varphi_i$. Then*

- *if $k \geq 3(n-m)$, then $\vec{\mathbf{c}}_n^m \not\models \varphi_k$ and*
- *if $k < (n-m)$, then $\vec{\mathbf{c}}_n^m \models \varphi_k$.*

Proof.

- If $m = n$, then $\vec{\mathbf{c}}_n^m$ does not result in $F_{ab}C$.

  Consider $\vec{\mathbf{c}}_n^m$ with $m < n$: $\vec{\mathbf{c}}_n^m = \vec{\mathbf{d}}_m.c \triangleright a.a \triangleright b.b \triangleright a\vec{\mathbf{d}}_{n-m}$, where $\vec{\mathbf{d}}_i$ contains i alternations $a \triangleright b.b \triangleright a$. In the $a \triangleright b$ call, a teaches b the secret C. But there is no way for a to know whether b already knew C before the call: $\vec{\mathbf{d}}_m.c \triangleright a.a \triangleright b.b \triangleright a\vec{\mathbf{d}}_{n-m} \sim_a \vec{\mathbf{d}}_m.c \triangleright a.c \triangleright b.a \triangleright b.b \triangleright a\vec{\mathbf{d}}_{n-m}$. Note that in $\vec{\mathbf{d}}_m.c \triangleright a.c \triangleright b.a \triangleright b.b \triangleright a\vec{\mathbf{d}}_{n-m}$, agent b cannot tell whether a tells him C in the call $a \triangleright b$, since he already knows C by that point (and the observance is α). So $\vec{\mathbf{d}}_m.c \triangleright a.c \triangleright b.a \triangleright b.b \triangleright a\vec{\mathbf{d}}_{n-m} \sim_b \vec{\mathbf{d}}_m.c \triangleright b.a \triangleright b.b \triangleright a\vec{\mathbf{d}}_{n-m}$. In that sequence, a cannot tell whether b learned C before or after $a \triangleright b$. So $\vec{\mathbf{d}}_m.c \triangleright b.a \triangleright b.b \triangleright a\vec{\mathbf{d}}_{n-m} \sim_a \vec{\mathbf{d}}_m.a \triangleright b.c \triangleright b.b \triangleright a\vec{\mathbf{d}}_{n-m}$.

  In summary, we have the following sequences of indistinguishabilities.

$$
\begin{array}{rcl}
\vec{\mathbf{c}}_n^m & = & \vec{\mathbf{d}}_m.c \triangleright a.a \triangleright b.b \triangleright a.\vec{\mathbf{d}}_{n-m} \\
& \sim_a & \vec{\mathbf{d}}_m.c \triangleright a.c \triangleright b.a \triangleright b.b \triangleright a.\vec{\mathbf{d}}_{n-m} \\
& \sim_b & \vec{\mathbf{d}}_m.c \triangleright b.a \triangleright b.b \triangleright a.\vec{\mathbf{d}}_{n-m} \\
& \sim_a & \vec{\mathbf{d}}_m.a \triangleright b.c \triangleright b.b \triangleright a.\vec{\mathbf{d}}_{n-m} \\
& \sim_b & \vec{\mathbf{d}}_m.a \triangleright b.c \triangleright b.c \triangleright a.b \triangleright a.\vec{\mathbf{d}}_{n-m} \\
& \sim_a & \vec{\mathbf{d}}_m.a \triangleright b.c \triangleright a.b \triangleright a.\vec{\mathbf{d}}_{n-m} \\
& \sim_b & \vec{\mathbf{d}}_m.a \triangleright b.b \triangleright a.c \triangleright a.\vec{\mathbf{d}}_{n-m} \\
& = & \vec{\mathbf{c}}_n^{m+1}
\end{array}
$$

  So $\vec{\mathbf{c}}_n^n$ can be reached from $\vec{\mathbf{c}}_n^m$ using $3(n-m)$ iterations of $\sim_a$ and $\sim_b$ steps. It follows that for $k \geq 3(n-m)$, $\vec{\mathbf{c}}_n^m \not\models \varphi_k$.

- In a call sequence $\vec{\mathbf{c}}_n^m$, both agents know when they first learned the secret C. Agent a is uncertain about when b thinks a learned C, and the other way around, but that uncertainty about when C was learned can be over at most one iteration of $a \triangleright b.b \triangleright a$. As such, if $k < n - m$, we have $\vec{\mathbf{c}}_n^m \models \varphi_k$.

□

Corollary 5.8. *If $\tau = (\bullet, \triangleright, \alpha)$ or $\tau = (\bullet, \triangleleft, \alpha)$, then there are infinitely long sequences of calls where no call is redundant.*

For these call types, we therefore bound the number of calls using the bisimilarity argument given in Section 5.1

Remark 5.9. *In [5] a number of open questions about gossip problems are introduced. One of these questions [5, Problem 3] is whether common knowledge reduces to nested knowledge. That is to say, if we let E^* stand for common knowledge and E for "everybody knows", whether there is a $k \in \mathbb{N}$ such that $E^*\varphi$ is equivalent to $E^k\varphi$. Lemma 5.7 answer this question in the negative, for the call types $(\bullet, \triangleright, \alpha)$ and $(\bullet, \triangleleft, \alpha)$.*

5.4 The Axiomatization

Using the bounds on non-redundant call sequences developed in the preceding subsections, we can define the privacy and observance axioms for the asynchronous case. We use two privacy axioms for this case, $\mathbf{Pri}_1(\bullet)$ and $\mathbf{Pri}_2(\bullet)$, shown in Table 6.

Privacy Axioms		
$\mathbf{Pri}_1(\bullet)$	$[\vec{\mathbf{c}}]K_a\varphi \leftrightarrow K_a\varphi$	$a \notin Ag(\vec{\mathbf{c}})$
$\mathbf{Pri}_2(\bullet)$	$K_a\varphi \rightarrow K_a[\vec{\mathbf{c}}]\varphi$	$a \notin Ag(\vec{\mathbf{c}})$

Table 6: Axioms for the the effect of privacy.

Together, the two privacy axioms represent the fact that an agent is completely unaware of any calls that they are not themselves involved in. The first axiom $\mathbf{Pri}_1(\bullet)$ states that any calls that a is not involved in do not affect a's knowledge. After all, if $a \notin Ag(\vec{\mathbf{c}})$ then a does not notice any of the calls $\vec{\mathbf{c}}$ taking place, so a knows φ after $\vec{\mathbf{c}}$ if and only if a already knew φ before the call sequence. In other words, if $a \notin Ag(\vec{\mathbf{c}})$ then $[\vec{\mathbf{c}}]K_a\varphi \leftrightarrow K_a\varphi$.

The second privacy axiom $\mathbf{Pri}_2(\bullet)$ is about an agent a reasoning about possible calls. For any call sequence $\vec{\mathbf{c}}$ not involving a, it is impossible for a to be certain that $\vec{\mathbf{c}}$ hasn't just happened. This means that in in order for a to know for certain that φ is true, a must also know that φ will remain true after $\vec{\mathbf{c}}$. In other words, if $a \notin Ag(\vec{\mathbf{c}})$ then $K_a\varphi \rightarrow K_a[\vec{\mathbf{c}}]\varphi$.

The observance axioms are a bit more complicated. Here, we use the bounds that we introduced in the preceding subsections. For $n \in \mathbb{N}$, let $\mathcal{E}_a(n) := \{\vec{\mathbf{c}} \mid \ell(\vec{\mathbf{c}}) \leq n \text{ and } \vec{\mathbf{c}} \sim_a \epsilon\}$. Furthermore, let $f(\varphi)$ be the number of n-bisimilarity classes, where n is the depth of φ. Then the observance axioms are as shown in Table 7.

If $\tau = (\bullet, \mathsf{d}, \mathsf{o})$, the proof system $\mathbf{G}^\tau$ consists of the basic axioms and rules from Tables 1, 2 and 3, together with the axioms $\mathbf{Pri}_1(\bullet)$ and $\mathbf{Pri}_2(\bullet)$ from Table 6 and the relevant observance axioms from Table 7.

Proposition 5.10. *The proof system $\mathbf{G}^\tau$ is sound for $\mathcal{L}^\tau$ on the class of all gossip models for call types $\tau = (\bullet, \mathsf{d}, \mathsf{o})$.*

Axioms for observance level β for sender ($\mathbf{Obs}^s$) and receiver ($\mathbf{Obs}r$)		
$\mathbf{Obs}^s(\bullet, \triangleright, \beta)$	$[a \triangleright b]K_a\varphi \leftrightarrow K_a \bigwedge_{\vec{\mathbf{c}} \in \mathcal{E}_a(2\|Ag\|^3)} [a \triangleright b.\vec{\mathbf{c}}]\varphi$	
$\mathbf{Obs}^r(\bullet, \triangleright, \beta)$	$[a \triangleright b]K_b\varphi \leftrightarrow \bigvee_{\mathsf{Q} \subseteq S}(O_a\mathsf{Q} \wedge K_b(O_a\mathsf{Q} \to \bigwedge_{\vec{\mathbf{c}} \in \mathcal{E}_a(2\|Ag\|^3)} [a \triangleright b.\vec{\mathbf{c}}]\varphi))$	
$\mathbf{Obs}^s(\bullet, \triangleleft, \beta)$	$[a \triangleleft b]K_b\varphi \leftrightarrow K_b \bigwedge_{\vec{\mathbf{c}} \in \mathcal{E}_a(2\|Ag\|^3)} [a \triangleleft b.\vec{\mathbf{c}}]\varphi$	
$\mathbf{Obs}^r(\bullet, \triangleleft, \beta)$	$[a \triangleleft b]K_a\varphi \leftrightarrow \bigvee_{\mathsf{Q} \subseteq S}(O_b\mathsf{Q} \wedge K_a(O_b\mathsf{Q} \to \bigwedge_{\vec{\mathbf{c}} \in \mathcal{E}_a(2\|Ag\|^3)} [a \triangleleft b.\vec{\mathbf{c}}]\varphi))$	
$\mathbf{Obs}^{s,r}(\bullet, \diamond, \beta)$	$[a \diamond b]K_c\varphi \leftrightarrow \bigvee_{\mathsf{Q},\mathsf{R} \subseteq S}(O_a\mathsf{Q} \wedge O_b\mathsf{R} \wedge K_c((O_a\mathsf{Q} \wedge O_b\mathsf{R}) \to \bigwedge_{\vec{\mathbf{c}} \in \mathcal{E}_a(2\|Ag\|^3)} [a \diamond b.\vec{\mathbf{c}}]\varphi))$	$c \in \{a, b\}$
Axioms for observance level α for sender ($\mathbf{Obs}^s$) and receiver ($\mathbf{Obs}^r$)		
$\mathbf{Obs}^s(\bullet, \triangleright, \alpha)$	$[a \triangleright b]K_a\varphi \leftrightarrow K_a \bigwedge_{\vec{\mathbf{c}} \in \mathcal{E}_a(f(\varphi))} [a \triangleright b.\vec{\mathbf{c}}]\varphi$	
$\mathbf{Obs}^r(\bullet, \triangleright, \alpha)$	$[a \triangleright b]K_b\varphi \leftrightarrow \bigvee_{\mathsf{Q} \subseteq S}(O_{ab}\mathsf{Q} \wedge K_b(O_{ab}\mathsf{Q} \to \bigwedge_{\vec{\mathbf{c}} \in \mathcal{E}_a(f(\varphi))} [a \triangleright b.\vec{\mathbf{c}}]\varphi))$	
$\mathbf{Obs}^s(\bullet, \triangleleft, \alpha)$	$[a \triangleleft b]K_b\varphi \leftrightarrow K_b \bigwedge_{\vec{\mathbf{c}} \in \mathcal{E}_a(f(\varphi))} [a \triangleleft b.\vec{\mathbf{c}}]\varphi$	
$\mathbf{Obs}^r(\bullet, \triangleleft, \alpha)$	$[a \triangleleft b]K_a\varphi \leftrightarrow \bigvee_{\mathsf{Q} \subseteq S}(O_{ab}\mathsf{Q} \wedge K_a(O_{ab}\mathsf{Q} \to \bigwedge_{\vec{\mathbf{c}} \in \mathcal{E}_a(f(\varphi))} [a \triangleleft b.\vec{\mathbf{c}}]\varphi))$	
$\mathbf{Obs}^{s,r}(\bullet, \diamond, \alpha)$	$[a \diamond b]K_c\varphi \leftrightarrow \bigvee_{\mathsf{Q} \subseteq S}(O_{ab}\mathsf{Q} \wedge K_c(O_{ab}\mathsf{Q} \to \bigwedge_{\vec{\mathbf{c}} \in \mathcal{E}_a(2\|Ag\|^3)} [a \diamond b.\vec{\mathbf{c}}]\varphi))$	$c \in \{a, b\}$

Table 7: Axioms about observance.

Proof. The proposition is proven in the same way as Proposition 4.9, except that we make use of the bounds introduced in Sections 5.1 and 5.2. We therefore do not give the proof in detail here. □

As with the other call types, the proof system $\mathbf{G}^\tau$ with $\tau = (\bullet, \mathsf{d}, \mathsf{o})$ has reduction axioms for the operator $[\mathsf{c}]$, so every formula is provably equivalent to a formula of $\mathcal{L}^\tau_{[\,]\text{-free}}$.

Theorem 5.11. *Let $\tau = (\bullet, \mathsf{d}, \mathsf{o})$. Then for every formula $\varphi \in \mathcal{L}\tau$ there is a formula $\psi \in \mathcal{L}^\tau_{[\,]\text{-free}}$ such that $\vdash^\tau \varphi \leftrightarrow \psi$.*

Unlike for the ○ and ◓ cases, completeness does not follow immediately. This is because, unlike the other privacy types, ● affects which formulas in $\mathcal{L}^\tau_{[\,]\text{-free}}$ are valid. The formula $K_a\neg F_bC$, for example, is not satisfiable for ●, since a can never be certain that no call bc or cb has happened. So the basic proof system $\mathbf{G}$ is not complete for for $\mathcal{L}^\tau_{[\,]\text{-free}}$ if $\tau = (\bullet, \mathsf{d}, \mathsf{o})$.

The solution lies in the axiom $\mathbf{Pri}_2(\bullet)$. The axiom states that $K_a\varphi \to K_a[\vec{\mathbf{c}}]\varphi$ if $a \notin Ag(\vec{\mathbf{c}}_2)$. This allows us to derive, for example, $\vdash \neg K_a\neg F_bC$, in the following way. First, using the appropriate effect axiom, we have $\vdash \neg[\mathsf{c}]\neg F_bC$, where $\mathsf{c} = bc$ or $\mathsf{c} = cb$, depending on the direction type. Then, using the contrapositive of $\mathbf{T}$, we get $\vdash \neg K_a[\mathsf{c}]\neg F_bC$. Finally, using the contrapositive of $\mathbf{Pri}_2(\bullet)$, we conclude that $\vdash \neg K_a\neg F_bC$. For every $\mathcal{L}^\tau_{[\,]\text{-free}}$ formula that is not satisfiable for privacy type ●, a similar proof exists.

Lemma 5.12. *Let $\tau = (\bullet, \mathsf{d}, \mathsf{o})$. Then $\mathbf{G}^\tau$ is sound and strongly complete for $\mathcal{L}^\tau_{[\,]\text{-free}}$ on the class of all gossip models.*

Completeness then follows immediately from Theorem 5.11 and Lemma 5.12.

Corollary 5.13. *The proof system $\mathbf{G}^\tau$ is strongly complete for $\mathcal{L}^\tau$ on the class of all gossip models for call type $\tau = (\bullet, \mathsf{d}, \mathsf{o})$.*

Together with Proposition 5.10, this yields the soundness and completeness that we desire.

Theorem 5.14. *The proof system $\mathbf{G}^\tau$ is sound and strongly complete for $\mathcal{L}^\tau$ on the class of all gossip models for call type $\tau = (\bullet, \mathsf{d}, \mathsf{o})$.*

6 Axiomatisation: the Tree Model

The proof system $\mathbf{G}^\tau$ is sound and complete with respect to the class of all gossip models. Unlike most other modal logics, however, our gossip logic has a single model that is of particular importance, namely the tree model. The axiomatizations that we introduced in the preceding sections are sound and complete for the class of all gossip models. So while those axiomatizations are interesting, we would additionally like to have axiomatizations that are sound and complete for the tree model specifically. We will do so by adding extra rules.

Recall that the axiomatization $\mathbf{G}^\tau$ reduces every formula to a call-free one. So it suffices to make our axiomatizations complete for $\mathcal{L}^\tau_{[\,]\text{-free}}$ with respect to the tree model. For the synchronous call types $\circ$ and $\mathbin{\rotatebox{90}{$\bullet$}}$ this is relatively easy.

In the origin world of the tree models for $(\circ, \mathsf{d}, \mathsf{o})$ and $(\mathbin{\rotatebox{90}{$\bullet$}}, \mathsf{d}, \mathsf{o})$ it is common knowledge among all agents that everyone only knows their own secret. The gossip states of the tree model are exactly those states that can be reached by some sequence of calls from this origin world.

Our language cannot express common knowledge, but we can express approximations of common knowledge. Recall that $root$ was defined as an abbreviation for the situation where every agents knows only their own secret. For $n \in \mathbb{N}$, we can then define $root^n$ recursively by

$$\begin{aligned} root^0 &:= root \\ root^{i+1} &:= root^i \wedge \bigwedge_{a \in Ag} K_a root^i \end{aligned}$$

We can then show that if $\varphi \in \mathcal{L}^\tau_{[\,]\text{-free}}$ is of depth at most n, then $root^n$ is a sufficiently close approximation of the tree model.

Lemma 6.1. *Let $(w, \vec{\mathbf{c}})$ be a gossip state such that $(w, \vec{\mathbf{c}}) \models root^n$, and let $\varphi \in \mathcal{L}^\tau_{[\,]\text{-free}}$ be a formula of depth at most n. Then for any call sequence $\vec{\mathbf{d}}$, we have $(w, \vec{\mathbf{c}}) \models [\vec{\mathbf{d}}]\varphi$ if and only if $(w_{root}, \epsilon) \models [\vec{\mathbf{d}}]\varphi$.*

Proof. If $(w, \vec{\mathbf{c}}) \models root^n$, then $(w, \vec{\mathbf{c}}) \leftrightsquigarrow_n (w_{root}, \epsilon)$. The lemma follows immediately from the facts that n-bisimilarity is preserved under calls and that truth of φ is preserved under n-bisimilarity. □

A formula is satisfiable in the tree model if and only if there is some $\vec{\mathbf{d}}$ such that $(w_{root}, \epsilon) \models [\vec{\mathbf{d}}]\varphi$. So, by the above lemma, φ is satisfiable if and only if there is some $\vec{\mathbf{d}}$ such that $\models root^n \to [\vec{\mathbf{d}}]\varphi$. The next step is to bound the length of the call sequence necessary to achieve φ if it is satisfiable. The general bound that we used in the preceding sections is still applicable, every satisfiable formula of depth n is satisfiable after a call sequence of length at most equal to the number of n-bisimilarity classes. But for some variants we can find better bounds.

First, consider the privacy type $\circ$. As discussed in Section 3, the tree model with that privacy type satisfies $\varphi \leftrightarrow K_a\varphi$ for every a. This implies that, in the tree model, two states cannot be distinguishable (by any formula) unless the are distinguishable by some atomic formula F_aB. The necessary number of calls to achieve any formula is therefore limited by the number of such atoms, i.e., by $|Ag|^2$.

Definition 6.2. *If $\tau = (\circ, \mathsf{d}, \mathsf{o})$, then the proof system $\mathbf{G}^{\tau}_{tree}$ consist of all rules and axioms of $\mathbf{G}^{\tau}$ plus the rule* **Tree***$(\circ)$ given by*

$$\textit{if } \vdash root^n \to [\vec{\mathbf{c}}]\varphi \textit{ for all } \vec{\mathbf{c}} \textit{ such that } \ell(\vec{\mathbf{c}}) \leq |Ag|^2 \textit{ then } \vdash \varphi$$

where n is the depth of φ.

Theorem 6.3. *If $\tau = (\circ, \mathsf{d}, \mathsf{o})$, then $\mathbf{G}^{\tau}_{tree}$ is sound and strongly complete for the tree model.*

For privacy type $\bullet\!\!\!\!-$, knowledge does not reduce to truth, i.e., $K_a\varphi$ is not equivalent to φ. The bound $|Ag|^2$ therefore fails, so we use the fallback option of bounding the call sequence length using the number of n-bisimilarity classes.

Definition 6.4. *If $\tau = (\bullet\!\!\!\!-, \mathsf{d}, \mathsf{o})$, then the proof system $\mathbf{G}^{\tau}_{tree}$ consists of all rules and axioms of $\mathbf{G}^{\tau}$ plus the rule* **Tree***$(\bullet\!\!\!\!-)$ given by*

$$\textit{if } \vdash root^n \to [\vec{\mathbf{c}}]\varphi \textit{ for all } \vec{\mathbf{c}} \textit{ such that } \ell(\vec{\mathbf{c}}) \leq m \textit{ then } \vdash \varphi$$

where n is the depth of φ and m is the number of n-bisimilarity equivalence classes.

Theorem 6.5. *If $\tau = (\bullet\!\!\!\!-, \mathsf{d}, \mathsf{o})$, then $\mathbf{G}^{\tau}_{tree}$ is sound and strongly complete for the tree model.*

This leaves us with the asynchronous privacy type $\bullet$. In that type, common knowledge of $root$ is impossible; even in the world w_{root} the agents consider it possible that $root$ is no longer true due to some call they were not involved in.

What we can do, is make use of the fact that the tree model only has one origin world w_{root}, so every state is of the form $(w_{root}, \vec{\mathbf{c}})$ for some $\vec{\mathbf{c}}$. The following proposition makes use of this property.

Definition 6.6. *The rules* $\mathbf{Tree}_1(\bullet)$, $\mathbf{Tree}_2(\bullet)$ *and* $\mathbf{Tree}_3(\bullet)$ *are given by*

$$\begin{array}{ll}
\mathbf{Tree}^{\infty}_1(\bullet) & \textit{If } \vdash root \to [\vec{\mathbf{c}}]\varphi \textit{ for all } (w_{root}, \vec{\mathbf{c}}) \sim_a (w_{root}, \vec{\mathbf{d}}), \\
& \textit{then } \vdash root \to [\vec{\mathbf{d}}]K_a\varphi. \\
\mathbf{Tree}^{\infty}_2(\bullet) & \textit{If } \vdash root \to [\vec{\mathbf{c}}]\varphi \textit{ for some } (w_{root}, \vec{\mathbf{c}}) \sim_a (w_{root}, \vec{\mathbf{d}}), \\
& \textit{then } \vdash root \to [\vec{\mathbf{d}}]\hat{K}_a\varphi. \\
\mathbf{Tree}^{\infty}_3(\bullet) & \textit{If } \vdash root \to [\vec{\mathbf{c}}]\varphi \textit{ for all } \vec{\mathbf{c}}, \textit{ then } \vdash \varphi.
\end{array}$$

Proposition 6.7. *The rules* $\mathbf{Tree}_1^\infty(\bullet)$, $\mathbf{Tree}_2^\infty(\bullet)$ *and* $\mathbf{Tree}_3^\infty(\bullet)$ *are sound with respect to the tree model.*

Proof. Soundness of the three rules follows from two facts: firstly, in the tree model there is only one world, namely w_{root}, and, secondly, (w_{root}, ϵ) is the only state where *root* holds. So suppose that $\models_{tree} root \to [\vec{\mathbf{c}}]\varphi$ for all $(w_{root}, \vec{\mathbf{c}}) \sim_a (w_{root}, \vec{\mathbf{d}})$. Then $(w_{root}, \epsilon) \models [\vec{\mathbf{c}}]\varphi$ for all such $\vec{\mathbf{c}}$ and therefore $(w_{root}, \vec{\mathbf{c}}) \models \varphi$ for all $\vec{\mathbf{c}}$ such that $(w_{root}, \vec{\mathbf{c}}) \sim_a (w_{root}, \vec{\mathbf{d}})$. Because w_{root} is the only world in this model, those are all the states that are a-indistinguishable from $(w_{root}, \vec{\mathbf{d}})$ so $(w_{root}, \vec{\mathbf{d}}) \models K_a\varphi$ and therefore $(w_{root}, \epsilon) \models [\vec{\mathbf{d}}]K_a\varphi$. Because (w_{root}, ϵ) is the only state where *root* holds, this implies that $\models_{tree} root \to [\vec{\mathbf{d}}]K_a\varphi$. We have now shown that $\mathbf{Tree}_1^\infty(\bullet)$ preserves validity, so it is sound.

Similarly, suppose that $\models root \to [\vec{\mathbf{c}}]\varphi$ for some $(w_{root}, \vec{\mathbf{c}}) \sim_a (w_{root}, \vec{\mathbf{d}})$. Then $(w_{root}, \vec{\mathbf{c}}) \models \varphi$ which, because of how $\vec{\mathbf{c}}$ was chosen, implies that $(w_{root}, \vec{\mathbf{d}}) \models \hat{K}_a\varphi$ and therefore $(w_{root}, \epsilon) \models [\vec{\mathbf{d}}]\hat{K}_a\varphi$. Again using the fact that (w_{root}, ϵ) is the only states where *root* is satisfied, we obtain $\models root \to [\vec{\mathbf{d}}]\hat{K}_a\varphi$. So the rule $\mathbf{Tree}_2^\infty(\bullet)$ also preserves validity, and is therefore sound.

Finally, soundness of $root \to [\vec{\mathbf{c}}]\varphi$ follows from the fact that every state in the tree model is of the form $(w_{root}, \vec{\mathbf{c}})$, so if $\neg\varphi$ were to hold in any state we would have $\not\models_{tree} root \to [\vec{\mathbf{c}}]\varphi$ for the appropriate call sequence $\vec{\mathbf{c}}$. □

While sound, the rules $\mathbf{Tree}_1^\infty(\bullet)$, $\mathbf{Tree}_2^\infty(\bullet)$ and $\mathbf{Tree}_3^\infty(\bullet)$ are not suitable for use in a recursive axiomatization, because they are infinitary, i.e., they have an infinite number of premises. Fortunately, the bounds that we introduced in the preceding sections still apply. We therefore define finitary variants of the three **Tree** rules as follows.

Definition 6.8. *For* $m \in \mathbb{N}$*, let the rules* $\mathbf{Tree}_1^m(\bullet)$, $\mathbf{Tree}_2^m(\bullet)$ *and* $\mathbf{Tree}_3^m(\bullet)$ *be given by*

$\mathbf{Tree}_1^m(\bullet)$ *if* $\vdash root \to [\vec{\mathbf{c}}]\varphi$ *for all* $\vec{\mathbf{c}}$ *such that* $\ell(\vec{\mathbf{c}}) \leq m \cdot (\ell(\vec{\mathbf{d}}) + 1)$ *and* $(w_{root}, \vec{\mathbf{c}}) \sim_a (w_{root}, \vec{\mathbf{d}})$ *then* $\vdash root \to [\vec{\mathbf{d}}]K_a\varphi$

$\mathbf{Tree}_1^m(\bullet)$ *if* $\vdash root \to [\vec{\mathbf{c}}]\varphi$ *for some* $\vec{\mathbf{c}}$ *such that* $\ell(\vec{\mathbf{c}}) \leq m \cdot (\ell(\vec{\mathbf{d}}) + 1)$ *and* $(w_{root}, \vec{\mathbf{c}}) \sim_a (w_{root}, \vec{\mathbf{d}})$ *then* $\vdash root \to [\vec{\mathbf{d}}]\hat{K}_a\varphi$

$\mathbf{Tree}_3^m(\bullet)$ *if* $\vdash root \to [\vec{\mathbf{c}}]\varphi$ *for all* $\vec{\mathbf{c}}$ *such that* $\ell(\vec{\mathbf{c}}) \leq m$ *then* $\vdash \varphi$

If $\tau = (\bullet, \diamond, \mathsf{o})$ *or* $\tau = (\bullet, \mathsf{d}, \beta)$*, let* $\mathbf{G}^\tau_{tree}$ *consist of the proof system* $\mathbf{G}^\tau$ *plus the rules* $\mathbf{Tree}_1^{2|Ag|^3}(\bullet)$, $\mathbf{Tree}_2^{2|Ag|^3}(\bullet)$ *and* $\mathbf{Tree}_3^{2|Ag|^3}(\bullet)$.

If $\tau = (\bullet, \triangleright, \alpha)$ *or* $\tau = (\bullet, \triangleleft, \alpha)$*, let* $\mathbf{G}^\tau_{tree}$ *consist of the proof system* $\mathbf{G}^\tau$ *plus the rules* $\mathbf{Tree}_1^{f(\varphi)}(\bullet)$, $\mathbf{Tree}_2^{f(\varphi)}(\bullet)$ *and* $\mathbf{Tree}_3^{f(\varphi)}(\bullet)$*, where* $f(\varphi)$ *is the number of* n*-bisimilarity equivalence classes and* n *is the depth of* φ*.*

Theorem 6.9. *If* $\tau = (\bullet, \mathsf{d}, \mathsf{o})$ *then* $\mathbf{G}^\tau_{tree}$ *is sound and strongly complete for the tree model.*

Proof. First, let use prove that the rules are sound. Using the bounds introduced in Sections 5.1 and 5.2, we can see that the premises of $\mathbf{Tree}_i^\infty(\bullet)$ (with $i \in \{1, 2, 3\}$)

are satisfied if and only if the premises of $\mathbf{Tree}_i^m(\bullet)$ are. By Proposition 6.7 it then follows that $\mathbf{G}^\tau_{tree}$ is sound for the tree model.

We continue by showing that $\mathbf{G}^\tau_{tree}$ is complete for the tree model. We start by showing that if $\models_{tree} \mathit{root} \to [\vec{\mathbf{d}}]\varphi$, then $\vdash \mathit{root} \to [\vec{\mathbf{d}}]\varphi$. Since every formula is provably equivalent to a $\mathcal{L}^\tau_{[\,]\text{-free}}$ formula, we can assume without loss of generality that $\varphi \in \mathcal{L}^\tau_{[\,]\text{-free}}$.

As base case, suppose that φ is a Boolean formula. Then $\mathit{root} \to [\vec{\mathbf{d}}]\varphi$ is already provable in $\mathbf{G}^\tau$, so in particular it is provable in $\mathbf{G}^\tau_{tree}$. Suppose then as induction hypothesis that the claim holds for all subformulas of φ. We continue by a case distinction on the main connective of φ, which we can assume without loss of generality to be K_a, $\hat{K}_a$, $\vee$ or $\wedge$.

First, suppose that $\varphi = K_a\psi$, so $\models_{tree} \mathit{root} \to [\vec{\mathbf{d}}]K_a\psi$. Then $(w_{root}, \epsilon) \models [\vec{\mathbf{d}}]K_a\psi$ and therefore $(w_{root}, \vec{\mathbf{d}}) \models K_a\psi$. It then follows that $(w_{root}, \vec{\mathbf{c}}) \models \psi$ for all $(w_{root}, \vec{\mathbf{c}}) \sim_a (w_{root}, \vec{\mathbf{d}})$. That, in turn, implies that $(w_{root}, \epsilon) \models [\vec{\mathbf{c}}]\psi$ for all such $\vec{\mathbf{c}}$. Because (w_{root}, ϵ) is the only state in the tree model where root holds, we get $\models_{tree} \mathit{root} \to [\vec{\mathbf{c}}]\psi$ for all $\vec{\mathbf{c}}$ such that $(w_{root}, \vec{\mathbf{c}}) \sim_a (w_{root}, \vec{\mathbf{d}})$. By the induction hypothesis, we obtain $\vdash \mathit{root} \to [\vec{\mathbf{c}}]\varphi$ for all such $\vec{\mathbf{c}}$. In particular, this means that $\vdash \mathit{root} \to [\vec{\mathbf{c}}]\varphi$ for every $\vec{\mathbf{c}}$ such that $\ell(\vec{\mathbf{c}}) \leq m{\cdot}(\ell(\vec{\mathbf{d}}){+}1)$ and $(w_{root}, \vec{\mathbf{c}}) \sim_a (w_{root}, \vec{\mathbf{d}})$, which by $\mathbf{Tree}_1^m(\bullet)$ yields $\vdash \mathit{root} \to [\vec{\mathbf{d}}]K_a\varphi$.

Secondly, suppose that $\varphi = \hat{K}_a\psi$, so $\models_{tree} \mathit{root} \to [\vec{\mathbf{d}}]\hat{K}_a\psi$. Then $(w_{root}, \vec{\mathbf{d}}) \models \hat{K}_a\psi$ and therefore $(w_{root}, \vec{\mathbf{c}}) \models \psi$ for some $(w_{root}, \vec{\mathbf{c}}) \sim_a (w_{root}, \vec{\mathbf{d}})$. Furthermore, we can choose this $\vec{\mathbf{c}}$ such that $\ell(\vec{\mathbf{c}}) \leq m \cdot (\ell(\vec{\mathbf{d}}) + 1)$. This implies that $\models_{tree} \mathit{root} \to [\vec{\mathbf{c}}]\psi$ and therefore, by the induction hypothesis, $\vdash \mathit{root} \to [\vec{\mathbf{c}}]\psi$. The rule $\mathbf{Tree}_2^m(\bullet)$ then yields $\vdash \mathit{root} \to [\vec{\mathbf{d}}]\hat{K}_a\varphi$.

Thirdly, suppose that $\varphi = \psi_1 \wedge \psi_2$, so $\models_{tree} \mathit{root} \to [\vec{\mathbf{d}}](\psi_1 \wedge \psi_2)$. Then $\models_{tree} \mathit{root} \to [\vec{\mathbf{d}}]\psi_1$ and $\models_{tree} \mathit{root} \to [\vec{\mathbf{d}}]\psi_2$, so by the induction hypothesis $\vdash \mathit{root} \to [\vec{\mathbf{d}}]\psi_1$ and $\vdash \mathit{root} \to [\vec{\mathbf{d}}]\psi_2$. It follows that $\vdash \mathit{root} \to [\vec{\mathbf{d}}](\psi_1 \wedge \psi_2)$.

Finally, suppose that $\varphi = \psi_1 \vee \psi_2$, so $\models_{tree} \mathit{root} \to [\vec{\mathbf{d}}](\psi_1 \vee \psi_2)$. This implies that $(w_{root}, \epsilon) \models [\vec{\mathbf{d}}](\psi_1 \vee \psi_2)$, which is equivalent to $(w_{root}, \epsilon) \models [\vec{\mathbf{d}}]\psi_1$ or $(w_{root}, \epsilon) \models [\vec{\mathbf{d}}]\psi_2$. Using the fact that $(\mathit{root}, \epsilon)$ is the only state in the rooted model where root holds, it follows that $\models_{tree} \mathit{root} \to [\vec{\mathbf{d}}]\psi_1$ or $\models_{tree} \mathit{root} \to [\vec{\mathbf{d}}]\psi_2$. By the induction hypothesis this yields $\vdash \mathit{root} \to [\vec{\mathbf{d}}]\psi_1$ or $\vdash \mathit{root} \to [\vec{\mathbf{d}}]\psi_2$, either of which can be used to obtain $\vdash \mathit{root} \to [\vec{\mathbf{d}}](\psi_1 \vee \psi_2)$.

This completes the proof that if $\models_{tree} \mathit{root} \to [\vec{\mathbf{d}}]\varphi$ then $\vdash \mathit{root} \to [\vec{\mathbf{d}}]\varphi$. Now, all that is left to do is to note that if $\models_{tree} \varphi$ then $\models_{tree} \mathit{root} \to [\vec{\mathbf{c}}]\varphi$ for every $\vec{\mathbf{c}}$. We have already shown that this implies that $\vdash \mathit{root} \to [\vec{\mathbf{c}}]\varphi$ for every $\vec{\mathbf{c}}$, and therefore in particular for ever $\vec{\mathbf{c}}$ such that $\ell(\vec{\mathbf{c}}) \leq m$. Using $\mathbf{Tree}_3^m(\bullet)$, this gives us $\vdash \varphi$.

□

7 Conclusion

Conclusion The *epistemic gossip problem* is a formal model of peer-to-peer communication, with so-called *calls* representing the communication itself and *secrets* representing the information that is exchanged. Different kinds of communication lead to slightly different models. This means that there are many different variants of the gossip problem, one for each possible way in which peer-to-peer communication can happen.

In this paper, we considered a large number of variants of the gossip problem, that differ on the extent to which communication is private (full observance, synchronous or asynchronous), the direction of information transfer (caller to callee, callee to caller, or both) and what information the agents gain about the set of secrets known by their communication partner (all secrets or only new secrets). We gave formal semantics for each of these variants, compared their properties, showed their model and validity checking problems to be decidable and introduced sound and complete axiomatizations for each of them. Importantly, the axiomatizations are reduction systems: every formula involving calls can be reduced to an equivalent formula that does not involve calls. This required proving bounds on the number of non-redundant calls in arbitrary sequences. Such bounds sometimes depend on the knowledge conditions that should hold after the execution of such sequences.

Our results that the logics are decidable imply that the termination of gossip protocols under all of the investigated variations is also decidable. Likewise, it is decidable whether a gossip protocol is successful, i.e., whether all agents will know all secrets once the protocol terminates. This is not immediately obvious, as gossip protocols allowing infinite call sequences cannot be formalized in our logical language. However, using our results for redundancy (with the direct knowledge bound where it is applicable and the n-bisimilarity bound in the two cases where the direct knowledge bound does not apply), we can identify any gossip protocol P with the finite set $\mathbf{C}_\mathsf{P}$ of its maximal non-redundant execution sequences. Termination then reduces to the validity of $\bigwedge_{\vec{\mathbf{c}}\in\mathbf{C}_\mathsf{P}}[\vec{\mathbf{c}}]\bigwedge_{a,b\in Ag}\neg\varphi_{ab}$ while success reduces to the validity of $\bigwedge_{\vec{\mathbf{c}}\in\mathbf{C}_\mathsf{P}}[\vec{\mathbf{c}}]\bigwedge_{a,b\in Ag}F_aB$.

Complexity and redundancy We showed that, in general, every formula φ of depth n that is satisfied after any call sequence is satisfied after a call sequence of length at most $f(n)$, where $f(n)$ is the number of n-bisimilarity classes. For certain variants, we also proved that there is a constant bound, i.e., one that does not depend on φ. Specifically, for $(\bullet,\diamond,\mathsf{o})$ and $(\bullet,\mathsf{d},\beta)$ we showed that a constant bound of $2|Ag|^3$ suffices. For the variants $(\bullet,\triangleright,\alpha)$ and $(\bullet,\triangleleft,\alpha)$ we showed that such a constant bound does not exist. For the observable case $(\circ,\mathsf{d},\mathsf{o})$ it is easy to see that a constant bound of $|Ag|^2$ suffices. This leaves only the synchronous privacy cases $(\text{◓},\mathsf{d},\mathsf{o})$. For those, existence of a constant bound is still an open question. Furthermore, for the cases where a constant bound exists it is not currently known whether the above bounds are tight.

We have used these bounds on the length of non-redundant call sequences to show that the model checking, tree validity checking and validity checking problems for all

variants of the gossip problem are decidable. Our decidability proofs are effective, in the sense that they prove decidability by providing a method for deciding them. We can therefore also find upper bounds of the complexity of the decision problems for each variant. Unfortunately, these bounds are rather high.

We can determine whether $M(I), (w, \vec{\mathbf{c}}) \models \varphi$ by first using the bound on the length of non-redundant call sequences to create a finite approximation of $M(I)$, and then using standard model checking techniques for modal logic to check whether φ holds in the approximation of $M(I)$. Determining whether $\models_{tree} \varphi$ can likewise be done by finding a finite approximation of M_{tree} and using standard model checking techniques to determine whether φ holds in that approximation. A simple, if rather naive, way to check whether $\models \varphi$ is to construct one initial model I per n-bisimilarity class, and then to check whether φ holds on each $M(I)$.

The constant bounds on the length of non-redundant call sequences are polynomial. In the variants of the gossip problem where these bound applies there are therefore at most exponentially many different non-redundant call sequences. The complexity of model checking and tree-validity checking for those variants is therefore at most exponential. In the variants of where the constant bounds do not apply we instead use the n-bisimilarity bound. That bound is non-elementary, however, so the bound on the complexity of model checking and tree validity checking for those variants is also non-elementary. Since we construct one initial model I for each n-bisimilarity class, the complexity bound on validity checking is non-elementary in every variant of the gossip problem.

None of these upper bounds have been shown to be tight. In fact, we suspect that most of them are not tight. Establishing tight upper bounds for the complexity of these decisions problems seems like an interesting avenue for future research.

Epistemic planning The relation between gossip protocols and epistemic planning was addressed in the already mentioned [16]. It seems to us that different gossip scenarios may possibly make for interesting case studies in epistemic planning, as their very restricted parameters allow for decidability in settings where general epistemic planning is undecidable. This may be in the context of temporal epistemic planning (as evidenced for gossip in the above [16] and also in the subsequent [18]) as well as in the context of dynamic epistemic planning.

Undecidability of epistemic planning has been a strong focus of recent research and has been obtained for very different settings [12, 8, 29, 15]. Among the sources of undecidability in epistemic planning are uncertainty about the actual action (observation) and thus unbounded (namely continuously expanding) model size while executing plans ([8]), but also the unrestricted modal depth of preconditions for plan execution, both for $\mathcal{S}5$ ([15]) and for $\mathcal{KD}45$ ([29]), resulting in unbounded models for another reason.

As in those more general kinds of epistemic planning, planning for gossip suffers from unbounded model size. What saves gossip planning from being undecidable is that both the effect of actions (calls) and the agents' ability to observe actions is highly structured and predictable. This is what allows us to reduce the unbounded models to finite approximations of those models, resulting in decidability.

Acknowledgments

We are grateful to Davide Grossi and Krzysztof Apt for their contributions to [2], which inspired this paper. Hans van Ditmarsch is also affiliated to IMSc, Chennai, as research associate. We would also like to thank the reviewers for their helpful comments.

References

[1] K. R. Apt, D. Grossi, and W. van der Hoek. Epistemic protocols for distributed gossiping. In *Proc. of 15th TARK*, pages 51–56. EPTCS, 2016.

[2] K. R. Apt, D. Grossi, and W. van der Hoek. When are two gossips the same? In G. Barthe, G. Sutcliffe, and M. Veanes, editors, *Proc. of 22nd LPAR*, volume 57 of *EPiC Series in Computing*, pages 36–55, 2018.

[3] K. R. Apt and D. Wojtczak. On the computational complexity of gossip protocols. In *Proceedings of IJCAI 2017*, pages 765–771, 2017.

[4] K. R. Apt and D. Wojtczak. Verification of distributed epistemic gossip protocols. *Journal of Artificial Intelligence Research*, 62:101–132, 2018.

[5] K. R. Apt and D. Wojtczak. Open problems in a logic of gossips. In *Proc. of 17th TARK*, pages 1–18. EPTCS volume 297, 2019.

[6] M. Attamah, H. van Ditmarsch, D. Grossi, and W. van der Hoek. Knowledge and gossip. In *Proceedings of ECAI'14*, pages 21–26. IOS Press, 2014.

[7] M. Attamah, H. van Ditmarsch, D. Grossi, and W. van der Hoek. The pleasure of gossip. In C. Başkent, L.S. Moss, and R. Ramanujam, editors, *Rohit Parikh on Logic, Language and Society*, pages 145–163. Springer, 2017.

[8] G. Aucher and T. Bolander. Undecidability in epistemic planning. In *Proc. of the 23rd IJCAI*, pages 27–33. AAAI Press, 2013.

[9] N. Bailey. *The Mathematical Theory of Epidemics*. Griffen Press, 1957.

[10] B. Baker and R. Shostak. Gossips and telephones. *Discrete Mathematics*, 2:197–193, 1972.

[11] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, 2001.

[12] T. Bolander and M.B. Andersen. Epistemic planning for single and multi-agent systems. *Journal of Applied Non-classical Logics*, 21(1):9–34, 2011.

[13] R. Bumby. A problem with telephones. *SIAM Journal of Algorithms and Discrete Methods*, 2:13–18, 1981.

[14] B. Chlebus and D. Kowalski. Robust gossiping with an application to consensus. *Journal of Computer and System Sciences*, 72:1262–1281, 2006.

[15] S. Lê Cong, S. Pinchinat, and F. Schwarzentruber. Small undecidable problems in epistemic planning. In J. Lang, editor, *Proc. of the 27th IJCAI*, pages 4780–4786, 2018.

[16] M. Cooper, A. Herzig, F. Maffre, F. Maris, and P. Régnier. Simple epistemic planning: Generalised gossiping. In *Proc. of ECAI*, pages 1563–1564, 2016.

[17] M. Cooper, A. Herzig, F. Maffre, F. Maris, and P. Régnier. The epistemic gossip problem. *Discrete Mathematics*, 342(3):654–663, 2019.

[18] M. Cooper, A. Herzig, F. Maris, and J. Vianey. Temporal epistemic gossip problems. In M. Slavkovik, editor, *Proc. of 16th EUMAS*, volume 11450 of *Lecture Notes in Computer Science*, pages 1–14, 2018.

[19] P. Th. Eugster, R. Guerraoui, A.-M. Kermarrec, and L. Massoulié. Epidemic information dissemination in distributed systems. *IEEE Computer*, 37(5):60–67, 2004.

[20] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning about knowledge*. The MIT Press, Cambridge, 1995.

[21] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. Knowledge-based programs. *Distributed Computing*, 10:199–225, 1997.

[22] P. Fraigniaud and E. Lazard. Methods and problems of communication in usual networks. *Discrete Applied Mathematics*, 53:79–133, 1994.

[23] A. Hajnal, E. C. Milner, and E. Szemeredi. A cure for the telephone disease. *Canadian Mathematical Bulletin*, 15:447–450, 1972.

[24] J. Halpern and L. Zuck. A little knowledge goes a long way: Knowledge-based derivations and correctness proofs for a family of protocols. *Journal of the ACM*, 39(3):449–478, 1992.

[25] S. M. Hedetniemi, S. T. Hedetniemi, and A. L. Liestman. A survey of gossiping and broadcasting in communication networks. *Networks*, 18(4):319–349, 1988.

[26] A. Herzig and F. Maffre. How to share knowledge by gossiping. *AI Communications*, 30(1):1–17, 2017.

[27] J. Hromkovic, R. Klasing, B. Monien, and R. Peine. Dissemination of information in interconnection networks (broadcasting and gossiping). In *Combinatorial Network Theory*, pages 125–212. Kluwer, 1996.

[28] J. Hromkovic, R. Klasing, A. Pelc, P. Ruzicka, and W. Unger. *Dissemination of Information in Communication Networks: Broadcasting, Gossiping, Leader Election, and Fault-Tolerance*. Springer, 2005.

[29] X. Huang, B. Fang, H. Wan, and Y. Liu. A general multi-agent epistemic planner based on higher-order belief change. In *Proc. of the 26th IJCAI*, pages 1093–1101, 2017.

[30] R. Kurki-Suonio. Towards programming with knowledge expressions. In *Proceedings of POPL'86*, pages 140–149, 1986.

[31] J.-J. Ch. Meyer and W. van der Hoek. *Epistemic Logic for AI and Computer Science*, volume 41 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 1995.

[32] J. Moon. Random exchanges of information. *Nieuw Archief voor Wiskunde*, 20:246–249, 1972.

[33] R. Parikh and R. Ramanujam. Distributed processing and the logic of knowledge. In *Logic of Programs*, LNCS 193, pages 256–268. Springer, 1985. Similar to *JoLLI* 12: 453–467, 2003.

[34] A. Procaccia, Y. Bachrach, and J. Rosenschein. Gossip-based aggregation of trust in decentralized reputation systems. In *Proceedings of IJCAI'07*, pages 1470–1475, 2007.

[35] Á. Seress. Quick gossiping without duplicate transmissions. *Graphs and Combinatorics*, 2:363–383, 1986.

[36] R. Tijdeman. On a telephone problem. *Nieuw Archief voor Wiskunde*, 3(XIX):188–192, 1971.

[37] H. van Ditmarsch, D. Grossi, A. Herzig, W. van der Hoek, and L. Kuijer. Parameters for epistemic gossip problems. In *Proceedings of LOFT'16*, 2016.

[38] H. van Ditmarsch, I. Kokkinis, and A. Stockmarr. Reachability and expectation in gossiping. In B. An, A. Bazzan, J. Leite, S. Villata, and L. van der Torre, editors, *Proc. of 20th PRIMA*, pages 93–109. Springer, 2017. LNCS 10621.

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

[40] H. van Ditmarsch, J. van Eijck, P. Pardo, R. Ramezanian, and F. Scharzentruber. Epistemic protocols for dynamic gossip. *Journal of Applied Logic*, 20:1–31, 2017.

[41] H. van Ditmarsch, J. van Eijck, P. Pardo, R. Ramezanian, and F. Schwarzentruber. Dynamic gossip. *Bulletin of the Iranian Mathematical Society*, 45(3):701–728, 2019.