# Gossip in Dynamic Networks

Hans van Ditmarsch, Jan van Eijck, Pere Pardo,
Rahim Ramezanian, François Schwarzentruber

Appeared in:

Liber Amicorum Alberti
A Tribute to Albert Visser
edited by Jan van Eijck, Rosalie Iemhoff and Joost J. Joosten
College Publications, London, 2016
pp 91–98

**Abstract**

A gossip protocol is a procedure for spreading secrets among a group of agents, using a connection graph. In this paper the problem of designing and analyzing gossip protocols is given a dynamic twist by assuming that when a call is established not only secrets are exchanged but also contact list, i.e., links in the gossip graph. Thus, each call in the gossip graph changes both the graph and the distribution of secrets. This paper gives a full characterization for the class of dynamic gossip graphs where the Learn New Secrets protocol (make a call to an agent if you know the number but not the secret of that agent) is successful.[1]

*For Albert Visser*

## 1 How to Spread Secrets

This contribution is offered to Albert Visser in the knowledge that the topic will delight him.[2] Gossip is idle talk about other people, and it typically involves

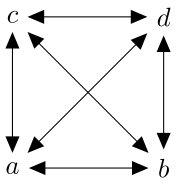
[1] This research was initiated when Jan van Eijck visited the other authors at LORIA in Nancy, in April 2015.

[2] Jan has many fond memories of interactions with Albert in the past: a joint talk on Montague grammar for the *Vereniging voor Logica*, the writing of *Inzien en Bewijzen*, running the Parallels Project together, and so on. Hans kindly remembers Albert from his early days in Utrecht as a mathematics and philosophy student, and from many other occasions such as the Alice in Wonderland workshop at the *Internationale School voor Wijsbegeerte*. Rahim's PhD-supervisor Mohammad Ardeshir recalls a memorable visit of Albert to Iran and their shared interests in intuitionistic logics.

details not confirmed as true. Not something that Albert engages in, but still connected to his interests in various ways. The formal study of how gossip spreads investigates the mechanisms behind the diffusion of information, and information and its growth are at the core of logic, from intuitionism to dynamic semantics for natural language.

Gossip protocols are procedures for spreading secrets among a group of agents, using a connection graph. It is assumed that everyone has a unique secret. The assumption that each agent starts out with a secret only known to that agent will enable us to trace each piece of information back to its unique source.

In the original set-up a totally connected graph was assumed. One of the key questions was to find a minimal sequence of calls to achieve a state where all agents knew all secrets. The assumption was that during a call, all secrets were exchanged. As it turns out, in a totally connected graph with $n > 3$ agents, $2n - 4$ calls are sufficient for this. Consider the totally connected graph with four agents.



A possible calling sequence for ensuring that all secrets get shared by everyone is $ab; cd; ac; bd$. If $e$ is also present, precede this sequence with $ae$, and close off with $ae$. Thus, in a network with four agents, all secrets can be shared in 6 calls. In general, two extra calls are sufficient for each additional agent, and we have that the number of calls for $n+1$ agents equals $2n - 4 + 2 = 2(n + 1) - 4$. It follows that $2n - 4$ calls are always enough. It is a bit trickier to show that $2n - 4$ calls are needed: see the original [Tij71], or [Hur00] and the references given there.

In the case above the gossip procedure is regulated by an outside authority, but in distributed computing we look for procedures that do not need outside regulation. A possible distributed protocol for gossip spreading could be:

> **Search For Secrets**
>
> While not every agent knows all secrets, let an agent $x$ who does not know all secrets randomly select an agent $y$, and let $x$ call $y$.

The Search for Secrets protocol has the advantage of simplicity, but note that it does not exclude redundant calls. Here is a distributed protocol (proposed in [AvDGvdH14]) that tries to avoid such redundancy.

> **Learn New Secrets**

> While not every agent knows all secrets, let an agent $x$ who does not know all secrets randomly select an agent $y$ such that $x$ does not know $y$'s secret, and let $x$ call $y$.

Note that the selection of the $x$ that makes the call still involves a minimal role for the environment: selecting the caller. We will assume that this selection is random.

There is a vast literature on gossiping and broadcasting in networks [HHL88], and there are connections with the study of the behaviour of epidemics [EGKM04]. Distributed gossip protocols are studied in [AvDGvdH14, AGvdH15]. In essence, these protocols investigate how information spreads through a network.

In this paper we give the problem of designing and analyzing (distributed) gossip protocols a dynamic twist by assuming that when a call is established not only secrets are exchanged but also contact lists. Thus, we drop the assumption that the graph of connections is complete from the start. Calls in the gossip graph are constrained by the current distribution of numbers, and each call changes both the graph and the distribution of secrets.

## 2 Gossip Graphs

Consider a finite set $A$ of agents, each with access to a set of other agents, and each carrying a unique secret. Then the access tables of the agents determine a graph.

Represent a graph $G$ with secrets (henceforth: gossip graph) as a triple $(A, N, S)$. $A$ is the (finite) set of vertices or agents, $N \subseteq A^2$ and $S \subseteq A^2$ are relations on $A$, with $Nxy$ expressing that $x$ has a link to $y$ (or: $x$ does know the contact details of $y$), and $Sxy$ expressing that $x$ does know the secret of $y$. Alternatively, we can think of $N$ and $S$ as functions in $A \to \mathcal{P}A$, so that $N_x$ is the set of agents whose numbers are known by $x$, and $S_x$ is the set of agents whose secrets are known by $x$.

In gossip graph $G = (A, N, S)$, an agent $x \in A$ is an expert if $S_x = A$, and if $B \subseteq A$, an agent $x \in A$ is a $B$-expert if $S_x \supseteq B$.

Represent a call from $x$ to $y$ as a tuple $xy$. The call $xy$ is possible in $G = (A, N, S)$ if $Nxy$. A call $xy$ merges the secret lists and the contact lists of $x$ and $y$. Let $G^{xy}$ be the result of this merge in $G$. That is, if $G = (A, N, S)$ and $x, y \in A$, then $G^{xy} = (A, N', S')$ where $N'_z = N_z$ for all $z \in A$ with $z \neq x, z \neq y$ and $N'_x = N'_y = N_x \cup N_y$, $S'_z = S_z$ for all $z \in A$ with $z \neq x, z \neq y$, and $S'_x = S'_y = S_x \cup S_y$. Alternatively, $N'$ can be given as $N \cup (\{(x, y), (y, x)\} \circ N)$, and $S'$ as $S \cup (\{(x, y), (y, x)\} \circ S)$.

A calling sequence $\sigma$ is a finite list of calls. We define the set $\mathbf{S}$ of calling sequences for agent set $A$ recursively as follows (assume $x, y$ range over $A$):

$$\sigma \quad ::= \quad \epsilon \mid \sigma; xy$$

where $\epsilon$ is the empty sequence.

If $\sigma, \tau \in \mathbf{S}$ we use $\sigma; \tau$ for the concatenation of $\sigma$ and $\tau$. Let $G^{\sigma} = (A, N^{\sigma}, S^{\sigma})$ be the graph that results after calling sequence $\sigma$. This is recursively defined as $G^{\epsilon} = G$, $G^{\sigma;xy} = (G^{\sigma})^{xy}$. We define possible calling sequences, as follows: $\epsilon$ is possible on any $G$, and $\sigma; xy$ is possible on $G$ iff $\sigma$ is possible on $G$, and $N^{\sigma}xy$ holds.

We say that $G = (A, N, S)$ has accessible secrets if $I_A \subseteq S \subseteq N$, where $I_A = \{(a, a) \mid a \in A\}$. Thus, $G$ has accessible secrets iff every agent knows her own secret and moreover, if agent $x$ knows the secret of $y$, $x$ also knows the number of $y$. Note if $G = (A, N, S)$ has accessible secrets, then $I_A \subseteq N$. This may look strange, as no agent is ever going to call itself, but one can also think if this as expressing the requirement that agents know their own number.

**Proposition 1.** *Let $G = (A, N, S)$, and let $\sigma$ be a possible calling sequence for $G$. If $G$ has accessible secrets then $G^{\sigma}$ has accessible secrets.*

*Proof.* Induction on $\sigma$, using the fact that it follows from $S^{\sigma}_x \subseteq N^{\sigma}_x$ and $S^{\sigma}_y \subseteq N^{\sigma}_y$ that $S^{\sigma}_x \cup S^{\sigma}_y \subseteq N^{\sigma}_x \cup N^{\sigma}_y$, and therefore $S^{\sigma;xy}_x = S^{\sigma;xy}_y \subseteq N^{\sigma;xy}_x = N^{\sigma;xy}_y$. □

**Proposition 2.** *Let $G = (A, N, S)$, and let $\sigma$ be a possible calling sequence for $G$. Then $N^{\sigma} \subseteq (N \cup N^{-1})^*$.*

*Proof.* Induction on $\sigma$. The base case is clear. For the inductive case, assume $N^{\sigma} \subseteq (N \cup N^{-1})^*$. Assume $\sigma; xy$ is a possible call for $G$. Then $(x, y) \in N^{\sigma}$. Notice that $N^{\sigma;xy} = N^{\sigma} \cup \{(x, y), (y, x)\} \circ N^{\sigma}$. We are done if we can show that $\{(x, y), (y, x)\} \circ N^{\sigma} \subseteq (N \cup N^{-1})^*$. From $(x, y) \in N^{\sigma}$, by ih, $(x, y) \in (N \cup N^{-1})^*$. Since $(N \cup N^{-1})^*$ is symmetric, also $(y, x) \in (N \cup N^{-1})^*$. Therefore $\{(x, y), (y, x)\} \subseteq (N \cup N^{-1})^*$. By induction hypothesis and relational reasoning, it follows from this that $\{(x, y), (y, x)\} \circ N^{\sigma} \subseteq (N \cup N^{-1})^* \circ (N \cup N^{-1})^* = (N \cup N^{-1})^*$. □

A gossip graph $G = (A, N, S)$ is weakly connected if for all $x, y \in A$ there is an $N \cup N^{-1}$-path from $x$ to $y$.

**Theorem 3.** *If $\sigma$ is a possible calling sequence for $G = (A, N, S)$, then $G$ is weakly connected iff $G^{\sigma}$ is weakly connected.*

*Proof.* Left to right is immediate. Right to left from Proposition 2. □

**Theorem 4.** *If $G = (A, N, S)$ satisfies $I_A = S \subseteq N$ and $\sigma$ is a possible calling sequence for $G$, then $S^{\sigma} \circ N \subseteq N^{\sigma}$.*

*Proof.* Induction on $\sigma$. For the base case we have to show that $S \circ N \subseteq N$. We have $S \circ N = I_A \circ N = N \subseteq N$.

For the induction step, let $\sigma$ be a possible calling sequence, and assume $S^\sigma \circ N \subseteq N^\sigma$. Let $xy$ be a possible call in $G^\sigma$.

Let $(a, b) \in S^{\sigma;xy} \circ N$. If $(a, b) \in S^\sigma \circ N$, then by the induction hypothesis, $(a, b) \in N^\sigma$, and hence by $N^\sigma \subseteq N^{\sigma;xy}$ we get that $(a, b) \in N^{\sigma;xy}$, and done.

If $(a, b) \in S^{\sigma;xy} \circ N$ and $(a, b) \notin S^\sigma \circ N$, then we may assume (wlog) that $a = x$ and that there is some $z$ with $S^{\sigma;xy}xz$, and $Nzb$.

From $S^{\sigma;xy}xz$ it follows that either $S^\sigma xz$ or $S^\sigma yz$ (either $x$ or $y$ knew the secret of $z$ before the call $xy$).

In the former case, we have $(x, b) \in S^\sigma \circ N$, and therefore by the induction hypothesis, $(x, b) \in N^\sigma$. In the latter case, we have $(y, b) \in S^\sigma \circ N$, and therefore by the induction hypothesis, $(y, b) \in N^\sigma$.

From $(x, b) \in N^\sigma$ or $(y, b) \in N^\sigma$ it follows by the definition of $N^{\sigma;xy}$ that $(x, b) \in N^{\sigma;xy}$, and done. □

A gossip graph $G = (A, N, S)$ is *complete* if it holds for all $x \in A$ that $S_x = A$. That is, a gossip graph is complete if all agents know all secrets.

A terminal point in $G = (A, N, S)$ is a point $x$ for which $N_x \subseteq \{x\}$. That is, a terminal point is an agent that knows at most her own number. The skin of a graph $G = (A, N, S)$ is the set $\{x \in A \mid N_x \subseteq \{x\}\}$ (the set of terminal points). Let $s(G)$ be the result of skinning graph $G$, i.e. removing all terminal points from $G$. That is, $s(G) = (B, N', S')$ where $B = \{x \in A \mid N_x - \{x\} \neq \emptyset\}$, $N' = N \cap B^2$, $S' = S \cap B^2$. Note that skinning a graph is not a closure operation: there are graphs with $s(s(G)) \neq s(G)$.

$N$ is strongly connected on $G = (A, N, S)$ if for any $x, y \in A$ there is an $N$-path from $x$ to $y$. Call $G = (A, N, S)$ strongly connected if $N$ is strongly connected on $G$.

The **Search For Secrets** protocol now takes the following shape. Note that the only change is the requirement that the caller has to know the number of the agent that gets called.

> **Search For Secrets**
>
> While not every agent knows all secrets, randomly select a pair $xy$ such that $Nxy$ and let $x$ call $y$.

In some cases, the dynamics can speed up the calling. A circle with five agents $a \longrightarrow b \longrightarrow c \longrightarrow d \longrightarrow e \longrightarrow a$ needs $2n - 3 = 7$ calls before everyone knows all secrets [HHL88], but in our dynamic approach 6 calls are sufficient: $ab; cd; ea; de; ac; bc$. This shows that old questions about minimum lengths of calling sequences can receive new answers in this dynamic setting.

# 3 Learn New Secrets

The following protocol is studied in [AvDGvdH14, AGvdH15] in the context of totally connected graphs.

> **Learn New Secret Protocol**
>
> While not every agent is an expert, let an agent $x$ that is not an expert randomly choose an agent $y$ from the list of agents for which $Nxy$ but not $Sxy$, and perform the call $xy$.

This is like **Learn New Secrets** from the introductory section, but with the extra requirement that the caller has to know the number of the agent that gets called.

We define LNS-permitted calling sequences, as follows: $\epsilon$ is LNS-permitted on any $G$, and $\sigma; xy$ is LNS-permitted on $G$ iff $\sigma$ is LNS-permitted on $G$ and $xy$ is LNS-permitted on $G^\sigma$. A calling sequence $\sigma$ is LNS-stuck on $G$ if $\sigma$ is LNS-permitted on $G$, $G^\sigma$ is not complete, and no call is LNS-permitted on $G^\sigma$.

Consider the spider-in-the-web example again. Trying out all the possible calling sequences reveal that they all get stuck, because of the fact that in no call $xy$ the caller learns a useful new number. That is, all calls $xy$ are such that if $x$ learns the number of $z$, then $x$ also learns the secret of $z$. In the example picture, the LNS permitted sequences are all the permutations of $ad; bd; cd$, and they all get stuck. So it makes sense to ask ourselves which graphs can be completed by some particular protocol.

It is straightforward to define and implement search algorithms for LNS-permitted calling sequences and LNS-stuck calling sequences [EG15].

The LNS protocol is successful on $G$ if either $G$ is complete, or there is an LNS-permitted call $xy$, and after any LNS-permitted call $xy$ the LNS protocol is successful on $G^{xy}$. It follows that LNS is successful on $G$ iff every sequence of LNS-permitted calls $\sigma$ results in a graph $G^\sigma$ that is complete, or is such that there is an LNS-permitted call, and after any LNS-permitted call $xy$, LNS is successful on $G^{\sigma;xy}$.

It follows from this definition that the LNS protocol is not successful on $G$ iff there is a calling sequence $\sigma$ that is LNS-stuck on $G$. This gives a straightforward algorithm for recognizing the gossip graphs where LNS is successful:

> **LNS gossip graph algorithm**
>
> Search for an LNS-stuck calling sequence in depth-first fashion, and declare success if no such calling sequence can be found [EG15].

A calling sequence $\sigma$ for $G$ is LNS-maximal if $\sigma$ is LNS-permitted for $G$, and no calls are LNS-permitted in $G^\sigma$. A calling sequence $\sigma$ for $G = (A, N, S)$ is LNS-maximal within $B \subseteq A$ if all calls in $\sigma$ are within $B$, $\sigma$ is LNS-permitted for $G$, and no calls within $B$ are LNS-permitted in $G^\sigma$.

**Proposition 5.** *If $\sigma$ is an LNS-maximal calling sequence for $G$, and $G$ has accessible secrets, then $S^\sigma = N^\sigma$.*

*Proof.* Let $G$ be a gossip graph with accessible secrets, and let there be $x, y$ with $N^\sigma xy$ and not $S^\sigma xy$. Then the call $xy$ is LNS-permitted in $G^\sigma$, and contradiction with the LNS-maximality of $\sigma$. This shows $N^\sigma \subseteq S^\sigma$. The property $S^\sigma \subseteq N^\sigma$ follows from the fact that $G$ has accessible secrets, and Proposition 1. Together, this gives $S^\sigma = N^\sigma$. □

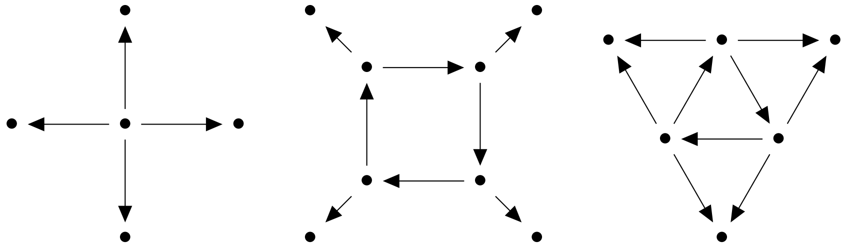
**Proposition 6.** *If $\sigma$ is an LNS-maximal calling sequence for $G$, and $G$ satisfies $I_A = S \subseteq N$, then $S^\sigma \circ N^* = S^\sigma$.*

*Proof.* $S^\sigma \subseteq S^\sigma \circ N^*$ by definition of $N^*$.

$S^\sigma \circ N^* \subseteq S^\sigma$: let $(x, y) \in S^\sigma \circ N^*$. Then for some $k \in \mathbb{N}$, $(x, y) \in S^\sigma \circ N^k$. We get from Theorem 4 plus Proposition 5 that $S^\sigma \circ N \subseteq S^\sigma$. Applying this fact $k$ times yields $(x, y) \in S^\sigma$. □

**Corollary 7.** *If $B$ is a strongly connected component of $G = (A, N, S)$ then any LNS-maximal calling sequence $\sigma$ within $B$ makes all elements of $B$ become experts for $B$.*

*Proof.* If $B$ is strongly connected, then $B \subseteq N^*$. If $\sigma$ is LNS-maximal, then this and Proposition 6 implies that $S^\sigma \circ B \subseteq S^\sigma$, which means that each member of $B$ has learnt the secret of all members of $B$. □



Call a graph $G = (A, N, S)$ a *sun* if $S = I_A \subseteq N$, $N$ is weakly connected on $G$, and $N$ is strongly connected on $s(G)$. The picture above gives three examples. We will show that $G$ is a sun if and only if LNS is successful on $G$.

**Theorem 8.** *The LNS protocol is successful for any sun $G$.*

*Proof.* Let $G = (A, N, S)$ be a sun. Let $\sigma$ be any LNS-maximal calling sequence for $G$. Let $x, y \in A$. We have to show that $S^\sigma xy$.

If $x$ is in the strongly connected core of $G$, then $N^* xy$. Because $Sxx$, also $S^\sigma xx$, and therefore $(x, y) \in S^\sigma \circ N^*$. By Proposition 6 it follows that $S^\sigma xy$.

If $x$ is a terminal node, then by maximality of $\sigma$, there is some $u$ with $(u, x) \in \sigma$. This means $N^\sigma uz$ for some $z$ with $Nzx$, for $u$ must have learnt $x$'s number from some such $z$. Thus, after the call $ux$, $x$ has the number of some $z$ with $Nzx$, that is, $N^\sigma xz$. By LNS-maximality of $\sigma$ it follows that $S^\sigma xz$. Since $z$ is in the strongly connected core of $G$, it follows that $(x, y) \in S^\sigma \circ N^*$. By Theorem 6, $S^\sigma xy$, and we are done. □

Let $\sim$ be the relation on $G = (A, N, S)$ given by $x \sim y$ iff there is an $N$-path from $x$ to $y$ and there is an $N$-path from $y$ to $x$. Then $\sim$ is an equivalence relation, and a cell in the partition induced by $\sim$ is called a strongly connected component of $G$. Use $[x]_\sim$ for the strongly connected component of $G$ that contains $x$. A strongly connected component $B$ is initial in $G$ if for all $x \in A$, $b \in B$: if $Nxb$ then $x \in B$. Notice that a gossip graph $G$ is strongly connected iff $\sim$ is universal on $G$.

**Theorem 9.** *Let $G$ be a connected graph with $I_A = S \subseteq N$. If $G$ is not a sun graph then the LNS protocol is not successful on $G$.*

*Proof.* Let $G = (A, N, S)$ be gossip graph that is weakly connected but not a sun graph. Let $H$ be an initial strongly connected component of $G$, and let $B$ be its carrier set.

There are $x, y, z \in A$ with $x \in B$, $y \notin B$, $z \notin B$, $Nxy$, and either $Nyz$ or $Nzy$. For if not, then $G$ is a sun. Notice that $B = [x]_\sim$.

Since $B$ is an initial strongly connected component, for all $u \in A - B$ and $v \in B$ we have $\neg Nuv$. In particular, $\neg Nzx$.

Let $\sigma'$ be an LNS-maximal calling sequence for $A - B$ in $G$. Then not $N^{\sigma'} zx$, for otherwise $\exists v \in A - B\ \exists w \in B$ with $Nvw$, and contradiction with the initiality of $B$. Let $\sigma''$ be an LNS-maximal calling sequence for $B$ in $G^{\sigma'}$. Then not $N^{\sigma';\sigma''} zx$, for calls in $\sigma''$ do not involve $z$. Let $By$ be a calling sequence where each member of $B$ calls $y$. Then not $N^{\sigma';\sigma'';By} zx$. Let $\sigma'''$ be an LNS-maximal calling sequence for $A - \{z\}$ in $G^{\sigma';\sigma'';By}$. Then not $N^{\sigma';\sigma'';By;\sigma'''} zx$.

Observe that $\sigma'; \sigma''; By; \sigma'''$ is an LNS maximal sequence of calls in $G$. For suppose $z$ could still make a call in $G^{\sigma';\sigma'';By;\sigma'''}$. Then the call cannot be within $A - B$, for otherwise contradiction with the fact that $\sigma'$ is maximal for $A - B$ in $G$. The call cannot be to an agent in $B$, for all calls made from or to $z$ are in $\sigma'$, and $z$ cannot have learnt a number in $B$ from these. Thus, $\sigma'; \sigma''; By; \sigma'''$ is LNS maximal. But $G^{\sigma';\sigma'';By;\sigma'''}$ is not complete, since not $N^{\sigma';\sigma'';By;\sigma'''} zx$. □

**Theorem 10.** *For any connected graph $G = (A, N, S)$ with $I_A = S \subseteq N$ the following holds: $s(G)$ is strongly connected iff the LNS protocol is successful for $G$.*

*Proof.* Immediate from Theorems 8 and 9. □

## 4 Further Questions

There are two ways in which the LNS protocol can be unsuccessful. On some graphs $G$ you can find successful LNS-permitted maximal call sequences if you are lucky, but you might get stuck if you change the order of the calls. For example, in the graph $a \longrightarrow b \longrightarrow c$ the calling sequences $ab; ac; bc$ and $ab; bc; ca$ are successful, but the calling sequence $bc; ab$ gets stuck. On the other hand, in the graph $a \longrightarrow b \longleftarrow c$, any calling sequence gets stuck: we have $ab; cb$ and stuck, and $cb; ab$ and stuck, and there are no success sequences. It would be of interest to characterize the graphs where the LNS protocol always gets stuck. This question is addressed in [DvEPetal16].

Also, it would be nice to characterize expected length of the calling sequences in this dynamic setting, for given protocols. For each protocol it makes sense to ask if it still holds that $2n - 4$ calls are enough. The new dynamic setting allows us to get new answers to old questions. What is the minimum, average, maximum number of calling sequences in a gossip graph with $n$ nodes, give property $X$ of the edge connection, and given that distributed protocol $P$ was used?

## Acknowledgement

We thank an anonymous referee for remarks and questions that have helped us improve the paper. We also acknowledge support from ERC project EPS 313360 and from CWI, Amsterdam.

## References

[AGvdH15] Krzyzstof R. Apt, Davide Grossi, and Wiebe van der Hoek. Epistemic protocols for distributed gossiping. In *Proceedings of TARK 2015*, 2015.

[AvDGvdH14] M. Attamah, H. van Ditmarsch, D. Grossi, and W. van der Hoek. Knowledge and gossip. In *Proc. of 21st ECAI*, pages 21–26. IOS Press, 2014.

[DvEPetal16] Hans Ditmarsch, Jan van Eijck, Pere Pardo, Rahim Ramezanian, and François Schwarzentruber. Dynamic gossip. Technical report, arxiv, 2016. `http://arxiv.org/abs/1511.00867`.

[EG15] Jan van Eijck and Malvin Gattinger. Gossip. Technical report, CWI, Amsterdam, available from `www.cwi.nl/~jve/papers/15/pdfs/Gossip.pdf`, 2015.

[EGKM04] Patrick Th. Eugster, Rachid Guerraoui, Anne-Marie Kermarrec, and Laurent Massoulié. Epidemic information dissemination in distributed systems. *IEEE Computer*, 37(5):60–67, 2004.

[HHL88] Sandra Mitchell Hedetniemi, Stephen T. Hedetniemi, and Arthur L. Liestman. A survey of gossiping and broadcasting in communication networks. *Networks*, 18(4):319–349, 1988.

[Hur00] C. A. J. Hurkens. Spreading gossip efficiently. *NAW*, 5(1):208–210, 2000.

[Tij71] R. Tijdeman. On a telephone problem. *Nieuw Archief voor Wiskunde*, 3(19):188–192, 1971.