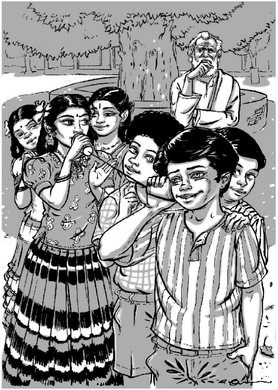
Gossip and Knowledge

Hans van Ditmarsch
Open University of the Netherlands

Malvin Gattinger
University of Amsterdam



presentation for ESSLLI Galway

August 2022

Friends exchanging secrets

Six friends each know a secret. They can call each other. In each call they exchange all the secrets they know. How many calls are needed for everyone to know all secrets?

Friends exchanging secrets

Six friends each know a secret. They can call each other. In each call they exchange all the secrets they know. How many calls are needed for everyone to know all secrets?

First consider four friends a, b, c, d who hold secrets A, B, C, D.
Four calls $ab;cd;ac;bd$ distribute all secrets. (AB is $\{A, B\}$, etc.)

$$A.B.C.D \xrightarrow{ab} AB.AB.C.D \xrightarrow{cd} AB.AB.CD.CD \xrightarrow{ac}$$
$$ABCD.AB.ABCD.CD \xrightarrow{bd} ABCD.ABCD.ABCD.ABCD$$

Now consider friends a, b, c, d, e, f with secrets A, B, C, D, E, F.
Eight calls $ae;af;ab;cd;ac;bd;ae;af$ distribute all secrets.
Minimum $2n - 4$ for $n \geq 4$.
[Tijdeman 1971; Baker & Shoshtak 1972; Hurkens 2000]

Friends exchanging secrets

> *Six friends each know a secret. They can call each other. In each call they exchange all the secrets they know. How many calls are needed for everyone to know all secrets?*

First consider four friends a, b, c, d who hold secrets A, B, C, D.
Four calls $ab; cd; ac; bd$ distribute all secrets. (AB is $\{A, B\}$, etc.)

$$A.B.C.D \overset{ab}{\rightarrow} AB.AB.C.D \overset{cd}{\rightarrow} AB.AB.CD.CD \overset{ac}{\rightarrow}$$
$$ABCD.AB.ABCD.CD \overset{bd}{\rightarrow} ABCD.ABCD.ABCD.ABCD$$

Now consider friends a, b, c, d, e, f with secrets A, B, C, D, E, F.
Eight calls $ae; af; ab; cd; ac; bd; ae; af$ distribute all secrets.
Minimum $2n - 4$ for $n \geq 4$.
[Tijdeman 1971; Baker & Shoshtak 1972; Hurkens 2000]

But how does c know that she should call d?
We want epistemic and distributed gossip protocols!

What do agents observe of calls not involving them?

Calls $ab;cd;ac;bd$ distribute all secrets. (All agents are experts.)
How does c know that she should call d?
What do agents observe of other calls? Let the agents be a, b, c, d.

- callers are observed: c notices a and b making a call: ab.
- calls are observed: c notices when two agents call: ab, ad, bd.
- time is observed: c notices when two agents **may** call: ab, ad, bd, ϵ.
- own calls are observed: c notices its own calls: $ab, ad, bd, \epsilon, ab;ad, ab;ad;bd, ab;ab;ab;ab, \ldots$

[Attamah *et al.*, *Knowledge and Gossip.* ECAI 2014]
[Apt *et al.*, *Epistemic Protocols for Distrib. Gossiping.* TARK 2015]

What do agents observe of calls involving them?

Let a know secrets X and let b know secrets Y.
What do they learn if they call each other? Different options:
— a learns b knew Y, b learns a knew X, a, b now know $X \cup Y$.
— a, b now know $X \cup Y$.
Under the first assumption they may learn more.
Consider $bc;ab;ad$ and $bc;ab;bd;ad$. Remove ad:

- After $bc;ab$, a knows ABC and d knows D.
- After $bc;ab;bd$, a knows ABC and d knows $ABCD$.
- So, after $bc;ab;ad$ and after $bc;ab;bd;ad$ a knows $ABCD$.
- The call sequences are indistinguishable for a.
- If a also learns what d knew before the call ad, a learns from $bc;ab;bd;ad$ that b or c must have called d.
- The call sequences are distinguishable for a.

Gossip protocol

Semantics of calls:

- If a knows secrets X and b knows secrets Y, then after call ab both know secrets $X \cup Y$.
- Define an equivalence relation $\sim_a$ between call sequences.
- a knows φ if φ is true after all equivalent call sequences.

Distributed Epistemic Gossip protocol:

- A gossip protocol is a program of shape "*until all agents know all secrets, choose agents x, y such that x knows that $\varphi(x, y)$, and let x call y.*" More distributed descriptions are possible.
- An execution sequence of a gossip protocol is successful if it terminates with all agents knowing all secrets.
- Strongly successful protocol: all executions are successful.
- Fairly successful protocol: all fair executions are successful.
- Weakly successful protocol: some executions are successful.
- Unsuccessful protocol: no executions are successful.

Examples of distributed epistemic gossip protocols

ANY
Until all agents know all secrets, any agent x calls any agent y.
LNS — Learn New Secrets
Until all agents know all secrets, an agent x calls an agent y whose secret it does not know.
KIG — Known Information Growth
Until all agents know all secrets, an agent x calls an agent y if x knows that x or y will learn a new secret in call xy.
PIG — Possible Information Growth
Until all agents know all secrets, an agent x calls an agent y if x considers possible that x or y will learn a new secret in call xy.

If only own calls are observed, LNS and KIG are identical.
If only own calls are observed, ANY and PIG are (nearly) identical.
[vD, van Eijck, Pardo, Ramezanian, Schwarzentruber: Epistemic protocols for dynamic gossip, JAL 2017]

Learn New Secrets protocol — example for four agents

LNS — Learn New Secrets

Until all agents know all secrets, an agent x calls an agent y whose secret it does not know.

Minimum execution length is $2n - 4$, maximum is $n(n-1)/2$.

4 agents:
A minimal call sequence $(2 \cdot 4 - 4 = 4)$ is *ab*;*cd*;*ac*;*bd*.
A maximal call sequence $(4(4-1)/2 = 6)$ is *ab*;*ac*;*ad*;*bc*;*bd*;*cd*.
There are also executions of length 5, e.g. *ab*;*ac*;*ad*;*bd*;*cd*.
In *ab*;*ac*;*ad*;*bc*;*bd*;*cd* call *cd* is the only LNS-permitted final call.
Not permitted but also achieving c learns D are *ac*, *bc*, *dc*, *ca*, *cb*.

[Attamah et al., ECAI 2014] [Haeupler, Journal of the ACM 2015] [Hedetniemi et al., Networks 1988] (NOHO)

Reachability

- What distributions of secrets are reachable by a call sequence? $AB.AB.C$, $ABC.ABC.ABC$, ...
- Distributions may be unreachable: $A.BC.C$, $AB.ABC.BC$.
- They may be reachable when calls are made in parallel.
- Reachability is modulo permutation of agent names: $(AB.AB.C)\binom{c}{a} = A.BC.BC$. (Isomorphic distributions)
- All distributions are subreachable: $A.BC.C$ is subreachable by agent d unknown to a calling c and then b.
- $ABCD.ABCD.ABC.ABD$ is reachable in ANY (any call) by $ab;ac;bd;ab$, but not in LNS.
- Reachability hierarchy for five epistemic gossip protocols.

[Gattinger: ILLC Dissertation Series DS-2018-11 (Ch. 6 Dynamic Gossip)]
[vD, Gattinger, Kuijer, Kokkinis: *Reachability of Five Gossip Protocols*. Workshop RP (Reachability Problems) 2019]

Expectation

- Gossip protocols with only finite executions need not be faster than gossip protocols with infinite execution sequences.
- Expectation of most distributed gossip protocols is $O(n \log n)$.
- This is the likelihood of a given agent to have called n agents if calls are random. (Coupon Collector problem)
- Variations such as LNS do not affect asymptotic behaviour except in a constant factor.
- By variations of ANY on partial networks $O(n \log^2 n)$ has been achieved [Haeupler, Giakkoupis, ...]. It is unclear whether epistemic gossip protocols can achieve that or improve that.
- Can the expectation be lowered when agents exchange more information than merely secrets?

[vD, Kokkinis, Stockmarr: *Reachability and Expectation in Gossiping*.]

Programming (few agents), computations (Markov chains), simulations.

Logic

- call sequences induce indist. relations to interpret knowledge
- calls are non-public events that correspond to action models
- In distributed gossip, call sequences of different length may be indistinguishable (asynchronous communication).
- In distributed gossip, a finite number of call sequences are, after all, indistinguishable for each agent: PDL-axiomatizable!

$$[ab]K_c\varphi \quad \leftrightarrow \quad (\dots) \bigwedge_{de\sim_c ab} K_c[de]\varphi \qquad \text{synchronous case}$$
$$[ab]K_c\varphi \quad \leftrightarrow \quad (\dots) \bigwedge_{\sigma\sim_c ab} K_c[\sigma]\varphi \qquad \text{asynchronous case}$$

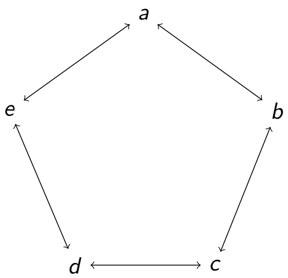
where σ is a call sequence of finite length.
For the synchronous case there are also DEL-axiomatizations.

[Attamah et al. ECAI 2014] [Apt, Wojtczak. JAIR 2018] [Gattinger. ILLC Diss. Series DS-2018-11] [vD, vd Hoek, Kuijer. The Logic of Gossiping, Artificial Intelligence Journal, 2020]

Gossip Graphs — when you can only call neighbours

We assumed that all agents can call all other agents.
Now assume that you can only call your neighbours: *gossip graph*.
This affects the optimal call sequence.
If the graph is connected, it is $2n - 3$ or $2n - 4$.

Example 5 nodes: 6 calls is not possible but 7 calls is possible.



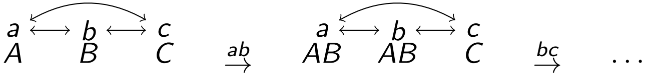
Dynamic Gossip: exchanging secrets and numbers

LNS — Learn New Secrets (Dynamic)
Until all agents know all secrets, an agent x calls an agent y whose number it knows and whose secret it does not know. (In a call, the callers exchange all secrets and all numbers they know.)

On fully connected graphs there is no difference.
Before, we displayed this as:

$$A.B.C \quad \xrightarrow{ab} \quad AB.AB.C \quad \xrightarrow{bc} \quad \ldots$$

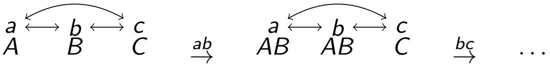
Now, we display this as:



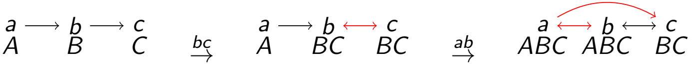
Dynamic Gossip — Learn New Secrets (with numbers)

Agents exchange all secrets and all numbers they know.

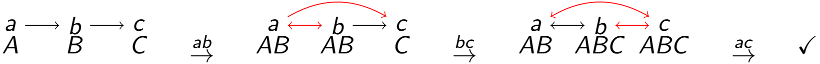
On fully connected graphs there is no difference.



On partially connected graphs deadlock is possible. (After bc;ab agent c cannot call agent a, because c does not have a's number.)



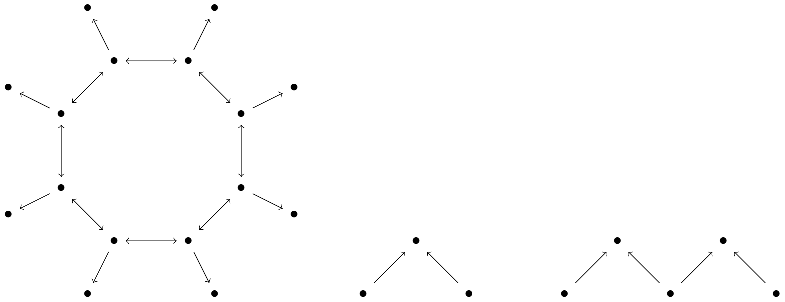
But sometimes deadlock can be avoided.
(After ab;bc agent a calls agent c.)



When exactly?

Dynamic Gossip — Characterization of LNS success

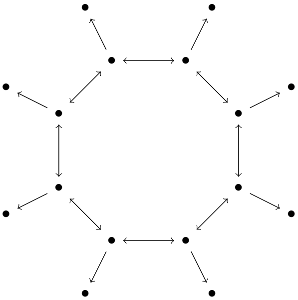
- If graph not weakly connected, unsuccessful. *Worse*:
  If graph a bush or double bush, unsuccessful.
- If graph strongly connected, strongly successful. *Better*:
  If graph a sun, strongly successful.



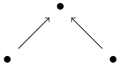
[vD, van Eijck, Pardo, Ramezanian, Schwarzentruber. *Dynamic Gossip*. Bulletin of the Iranian Mathematical Society, 2019]

Dynamic Gossip — Characterization of LNS success

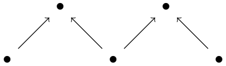
- A sun is a strongly connected graph to which may be linked maximal nodes.
- A bush is a tree with branching factor in the root at least 2.
- A double bush consists of two bushes joined in a leaf linked to their roots.



sun



bush

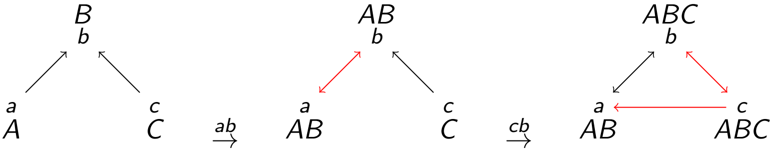


double bush

Dynamic Gossip — Characterization for LNS

- *LNS is unsuccessful on a weakly connected gossip graph*
  iff
  the weakly connected gossip graph is a bush or a double bush

For example, LNS is unsuccessful on this bush:



The (rather) hard part of the proof is to construct a successful call sequence on an arbitrary weakly connected graph that is not a bush or a double bush.

[vD *et al.*, *Dynamic Gossip*. BIMS 2019]

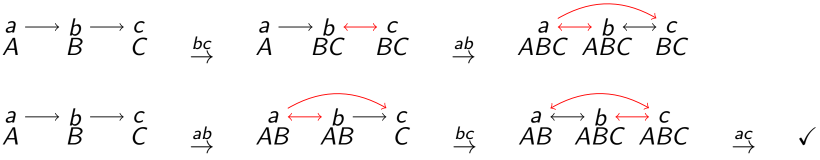
Common Knowledge and Gossip

What may or may not be common knowledge:

- The protocol
- The gossip graph (i.e., the network topology)
- The number of agents (i.e., nodes)
- That agents initially hold a single secret
- The time
- ...

These conditions are often implicit.

Common Knowledge and Gossip — strengthening LNS

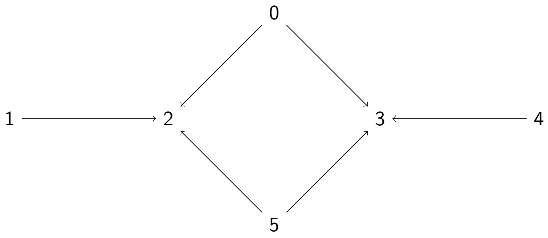


- LNS is weakly successful on $a \longrightarrow b \longrightarrow c$: if b calls first we get stuck, but if a calls first any extension is successful.
- We can *strengthen* LNS on this graph to ensure strong success instead of weak success, in different ways:
- LNS^{+} is strongly successful: after σ, a calls b if a knows the number but not the secret of b and noone knows a's number.
- $\text{LNS}^{\Diamond}$ is strongly successful: after σ, a calls b if a knows the number but not the secret of b and knows there is a successful LNS extension of $\sigma;ab$.
  (Assumes common knowledge of LNS and the gossip graph.)

[vD, Gattinger, Kuijer, Pardo. *Strengthening Gossip Protocols* . . . FLAP (IfCoLog Journal of Logics and their Applications) 2019]

Common Knowledge and Gossip [vD *et al.*, FLAP 2019]

Theorem: The protocol LNS **cannot** be strengthened such that it becomes strongly successful (i.e., on all gossip graphs).



LNS is weakly successful on this gossip graph, but there is no epistemic symmetric protocol that is a strengthening of LNS and that is strongly successful on it. It can be shown that 0 or 5 must make the first call. If 0 were to call 2, then 1 must now call 2. But any of 4 calls $02, 03, 52, 53$ is equally likely for 1. A successful sequence is 02;12;53;43;13;03;23;52;42.

Everyone knows that everyone knows all secrets

— $E\ All$: everyone knows all secrets.
— $E^2 All$: everyone knows that everyone knows all secrets.

LNS variation obtaining $E^2 All$:

Until a knows that all know all secrets: if a doesn't know all secrets, then a calls a b whose secret a does not know, else a calls a b who may not know all secrets.

Example for four agents:

ab;*cd*;*ac*;*bd*;	all agents know all secrets
ab;*ad*;	agent *a* knows that all agents know all secrets
bc;	agent *b* knows that all agents know all secrets
cd;	agents c, d know that all agents know all secrets

If agents only communicate secrets, $E^2 All$ is all they can get.
[vD, Gattinger, WoLLIC 22]: $E^3 All$ unsatisfiable, asynchronous

What if they can communicate more?

Everyone knows that everyone knows

To obtain $E^2 All$ we need $O(n^2)$ calls. We recall:

$ab; cd; ac; bd;$	all agents know all secrets
$ab; ad;$	agent a knows that all agents know all secrets
$bc;$	agent b knows that all agents know all secrets
$cd;$	agents c, d know that all agents know all secrets

Also communicating knowledge, to obtain $E^2 All$ we need $O(n)$ calls.

$ab; cd; ac; bd;$	all agents know all secrets
$ab;$	agent a informs b that a, c know all secrets
	agent b informs a that b, d know all secrets
	agents a, b know that all agents know all secrets
cd	agent c informs d that a, c know all secrets
	agent d informs c that b, d know all secrets
	agents c, d know that all agents know all secrets

[Cooper et al. *The epistemic gossip problem.* Discrete Math. 2019]

Full information protocols achieve arbitrary epistemic depth! To do.

Everyone knows that everyone knows

[Cooper et al.] obtain $E^k All$ with $(k+1)(n-2)$ calls.
This is an optimal but not a **distributed** scheduling of calls.

Although the (optimal) expectation with goal $E^2 All$ is $O(n)$, we recall that the expectation of many (distributed) gossip protocols with goal $E\ All$ is $O(n \log n)$. It is not known what the expectation is with goal $E^2 All$: $O(n \log n)$ or $O(n^2)$?

Consider goal $E^2 All$, agents only exchange secrets, and:
— agents knowing all agents know all secrets no longer answer calls;
— agents knowing all agents know all secrets no longer make calls.
Using the semantics for strengthening gossip protocols, there are strongly successful protocols **with goal** $E^2 All$. If synchronous, some reach **common knowledge** that everyone knows all secrets.
[vD, Gattinger, Ramezanian, *Everyone knows that everyone knows*]
— $E^3 All$ is unsatisfiable (given asynchrony & make any call)
[vD, Gattinger, *The Limits to Gossip (...)*. WoLLIC 2022]

Lying, deceit, and error

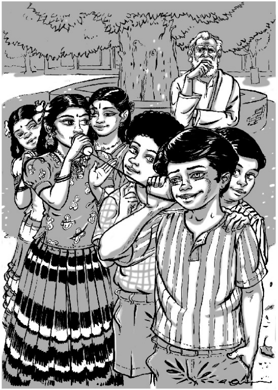
In the networks community, fault tolerant gossip protocols are big. Modelling lying, deceit and error in epistemic gossip protocols is somewhat uncharted territory.

— van den Berg: Unreliable Gossip. ILLC MoL-2018-01
— van den Berg, Gattinger:
Dealing with Unreliable Agents in Dynamic Gossip. DaLí 2020.

There seems ample scope to consider gossip protocols with a given maximum of faulty agents or faulty messages, analogous to Ulam games, and to determine optimal schedules under those conditions.

Further research

- Building bridges to the networks community
- Building bridges to the distributed computing community
- Strengthening the logic community



References [& more by those authors]

- Attamah, vD, Grossi, vdH: *Knowledge and Gossip*. ECAI 2014
- Apt, Grossi, van der Hoek: *Epistemic Protocols for Distributed Gossiping*. TARK 2015.
- vD, van Eijck, Pardo, Ramezanian, Schwarzentruber: *Dynamic Gossip*. BIMS 2019.
- Herzig, Maffre: *How to share knowledge by gossiping*. AI Commun. 2017.
- Cooper, Herzig, Maffre, Maris, Régnier. *The epistemic gossip problem.* Discrete Mathematics 2019.
- vD, Kokkinis, Stockmarr: *Reachability and Expectation in Gossiping*. PRIMA 2017.
- Apt, Wojtczak: *Verification of Distributed Epistemic Gossip Protocols*. JAIR 2018.
- vD, Gattinger, Kuijer, Pardo: *Strengthening Gossip Protocols using Protocol-Dependent Knowledge*. FLAP 2019.