From Coordination to Blockchain: The Elusive Trail of Common Knowledge

Joe Halpern
Cornell University

Includes joint work with Ron Fagin, Yoram Moses, Rafael Pass, and Moshe Vardi

The muddy children puzzle

The "classical" model of knowledge

The *possible worlds* model (over 50 years old!):

- Besides the actual state of affairs, an agent considers a number of other states of affairs to be possible.
- An agent *knows* a fact p if p is true in all the states of affairs, or worlds, that he thinks possible.

Knowledge in distributed systems

- A distributed system consists of a collection of processes connected by a communication network
- Each process has a local state (depending on the initial state, messages received, etc.).
- The *global state* of the system is a tuple consisting of each process' (local) state.
- A *run* of the system is a complete description of the system over time: formally, a function from times to global states.
- A *system* is a set of runs
- A protocol is described by a system: each run describes one possible execution of the protocol. At time m in run r, the system is in some global state.
- An agent *knows* a fact p at some point (r, m) if p is true at all the points (r', m') it considers possible; i.e., at all the points (r', m') where it is in the same local state.
- We write $(\mathcal{R}, r, m) \models \phi$ if the formula ϕ is true at the point (r, m) in the system $\mathcal{R}$.

Sound and complete axiomatizations

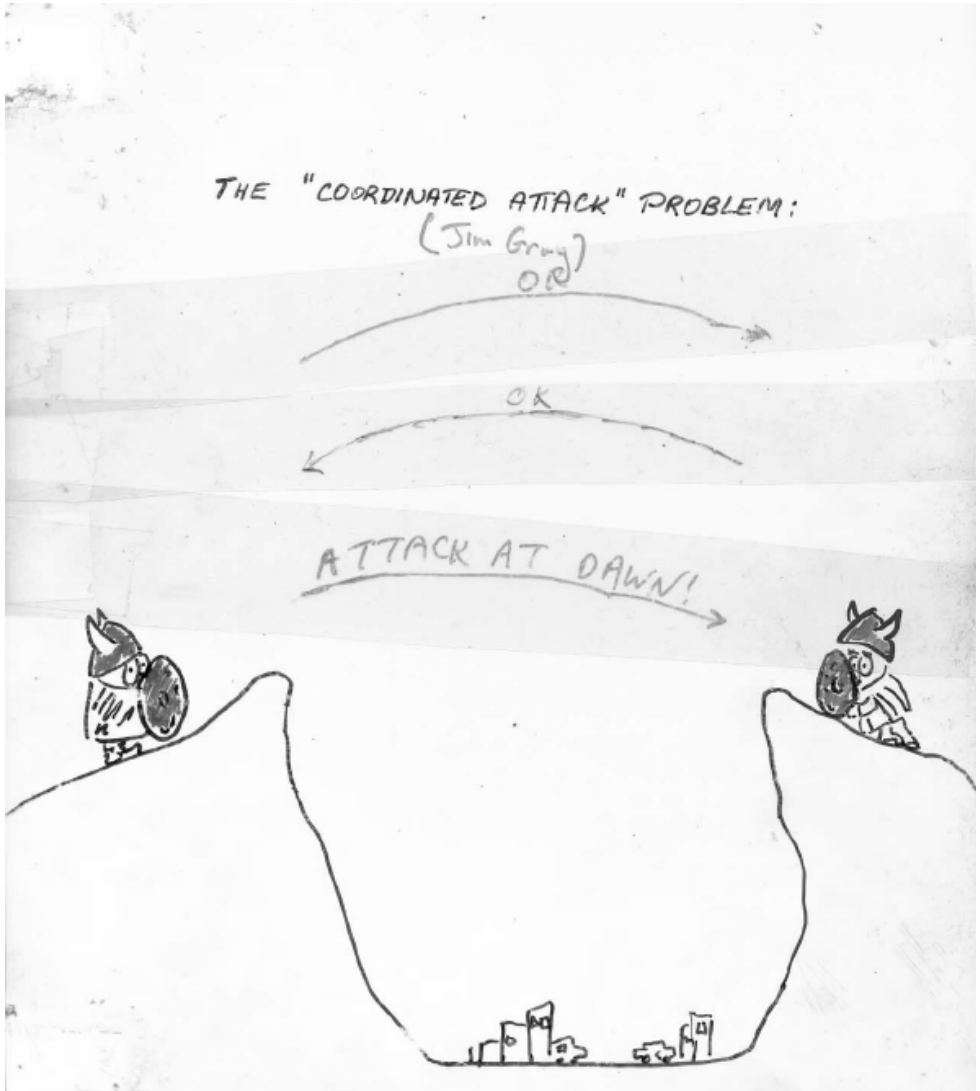
There are standard axiom systems for knowledge;

- Given our assumptions, we get a multi-agent version of the standard modal logic S5
- Key axioms:
  - $K_i\phi \Rightarrow \phi$ [If you know it, it's true]
  - $K_i\phi \Rightarrow K_iK_i\phi$ [You know what you know]
  - $\neg K_i\phi \Rightarrow K_i\neg K_i\phi$ [. . . and what you don't know]
  - $(K_i(\phi \Rightarrow \psi) \wedge K_i\phi) \Rightarrow K_i\psi$ [Closure under logical consequence]

What about common knowledge? Again, there are standard axioms and inference rules:

- From $\phi \Rightarrow E(\phi \wedge \psi)$ infer $\phi \Rightarrow C\psi$ [Induction rule]
- $C\phi \Leftrightarrow EC\phi$ [Fixed point axiom]

The coordinated attack problem



Each time the messenger makes it, the level of knowledge rises.
Let $m =$ "General R sent a message saying 'attack at dawn' "
First $K_G m$, then $K_R K_G m$, $K_G K_R K_G m$, ...

Proposition: (Halpern-Moses) m will never become common knowledge using a k-round handshake protocol.

Theorem: m will never become common knowledge in any run of any protocol. In fact, common knowledge is not attainable in any system where communication is not guaranteed.
But what about coordinated attack?

Each time the messenger makes it, the level of knowledge rises.
Let $m =$ "General R sent a message saying 'attack at dawn' "
First $K_G m$, then $K_R K_G m$, $K_G K_R K_G m$, ...

Proposition: (Halpern-Moses) m will never become common knowledge using a k-round handshake protocol.

Theorem: m will never become common knowledge in any run of any protocol. In fact, common knowledge is not attainable in any system where communication is not guaranteed.
But what about coordinated attack?

Agreement implies common knowledge.

Corollary: Any protocol that guarantees that if one of the generals attacks, then the other does so at the same time, is a protocol where necessarily neither general attacks.

(Provided we assume that in the absence of messages, neither general will attack.)

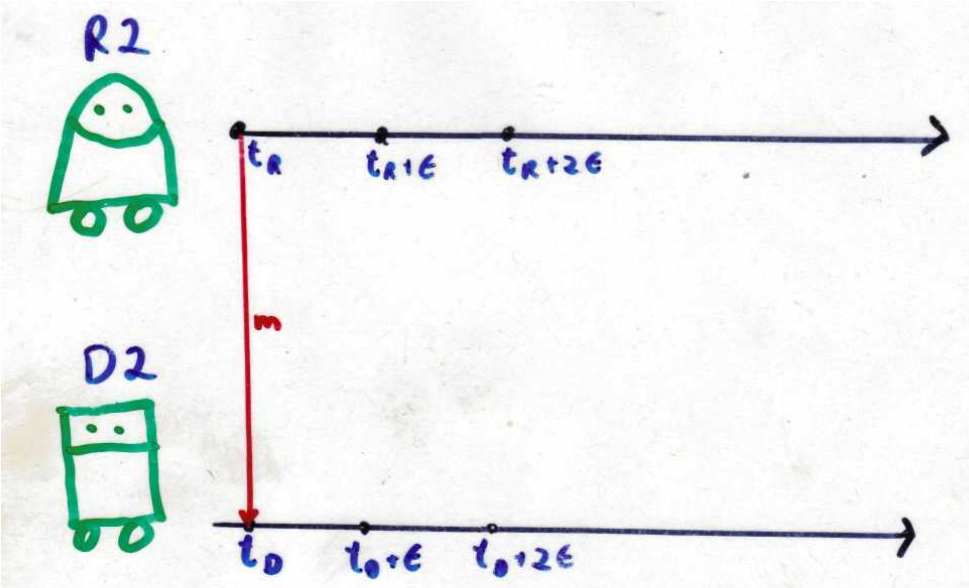
We have shown that common knowledge is not attainable if communication is not guaranteed.

We can easily show that common knowledge is also not attainable if communication is guaranteed, but there is no upper bound on message delivery time.

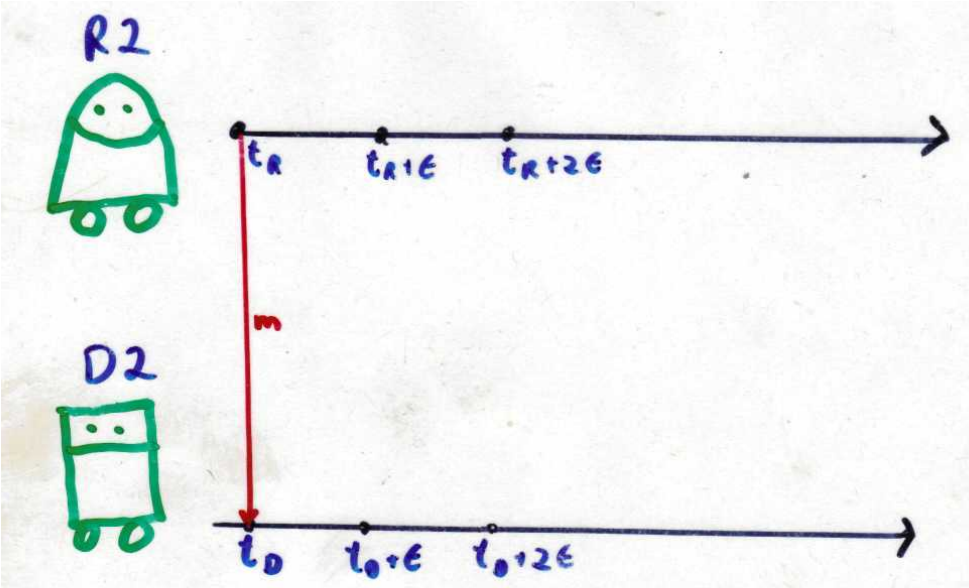
We have shown that common knowledge is not attainable if communication is not guaranteed.

We can easily show that common knowledge is also not attainable if communication is guaranteed, but there is no upper bound on message delivery time.
What if there is an upper bound on message delivery time, but the actual message delivery time is uncertain?

Suppose we have an upper bound of ϵ, but messages might take anywhere from 0 to ϵ to arrive:

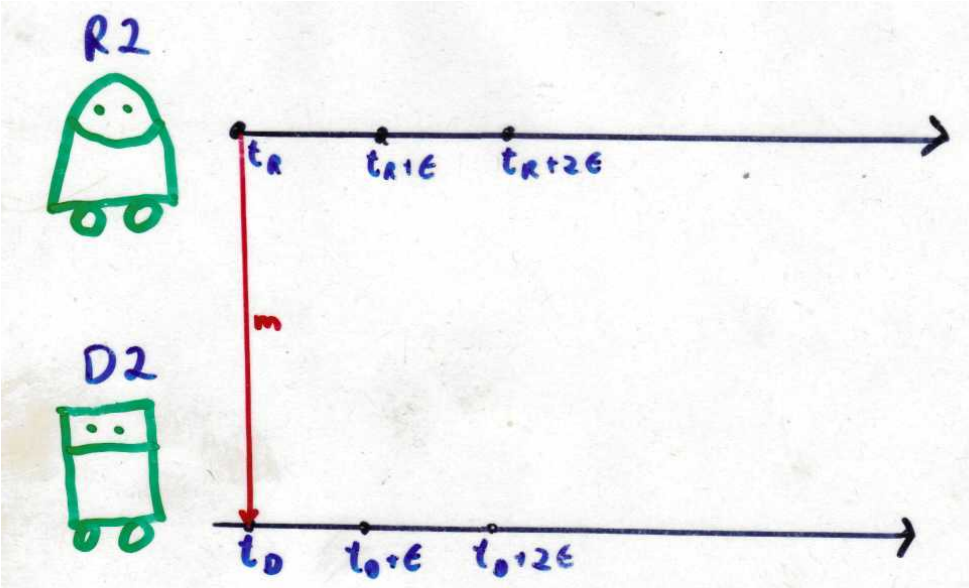


Suppose we have an upper bound of ϵ, but messages might take anywhere from 0 to ϵ to arrive:



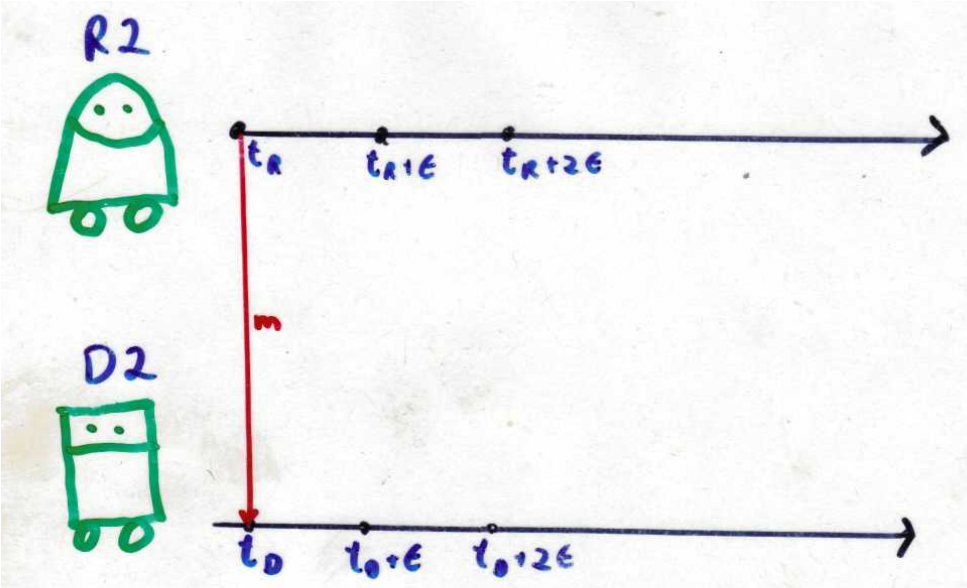
- At time $t_R + \epsilon$, have $K_R K_D m$

Suppose we have an upper bound of ϵ, but messages might take anywhere from 0 to ϵ to arrive:



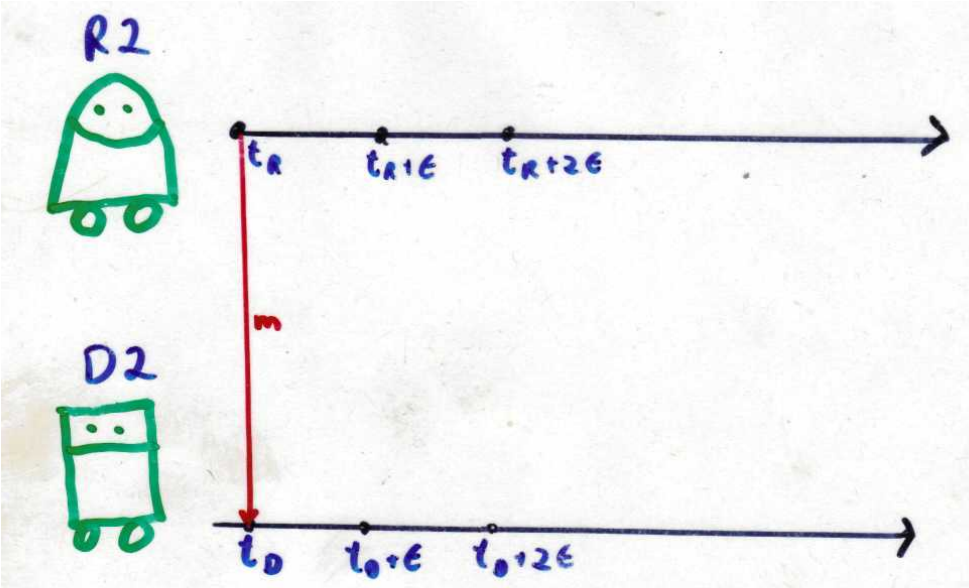
- At time $t_R + \epsilon$, have $K_R K_D m$
- At time $t_D + \epsilon$, have $K_D K_R K_D m$

Suppose we have an upper bound of ϵ, but messages might take anywhere from 0 to ϵ to arrive:



- At time $t_R + \epsilon$, have $K_R K_D m$
- At time $t_D + \epsilon$, have $K_D K_R K_D m$
- At time $t_R + 2\epsilon$, have $K_R K_D K_R K_D m$
- ...

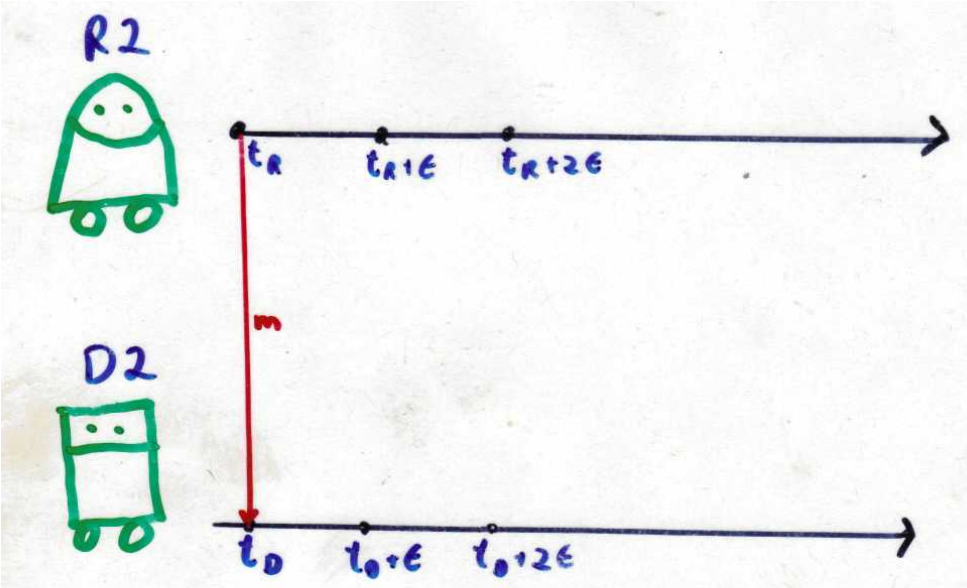
Suppose we have an upper bound of ϵ, but messages might take anywhere from 0 to ϵ to arrive:



- At time $t_R + \epsilon$, have $K_R K_D m$
- At time $t_D + \epsilon$, have $K_D K_R K_D m$
- At time $t_R + 2\epsilon$, have $K_R K_D K_R K_D m$
- ...

What about Cm?

Suppose we have an upper bound of ϵ, but messages might take anywhere from 0 to ϵ to arrive:



- At time $t_R + \epsilon$, have $K_R K_D m$
- At time $t_D + \epsilon$, have $K_D K_R K_D m$
- At time $t_R + 2\epsilon$, have $K_R K_D K_R K_D m$
- ...

What about Cm?

- Never!

The situation is very different if there is a global clock and the messages are timestamped:
If R2 says "m; the time is 5 P.M.", this message becomes common knowledge at $5 + \epsilon$.
Theorem: Common knowledge requires synchronized clocks.
Corollary: In any system where message delivery time is uncertain and clocks are not initially synchronized, common knowledge is not attainable.

Conclusion?

Although common knowledge is a desirable and consistent state of knowledge, it is not attainable in practical systems.

Weaker (attainable) variants of common knowledge

- Epsilon common knowledge:

$$C^{\epsilon}p \equiv \bigcirc^{\epsilon} Ep \wedge \bigcirc^{\epsilon} E \bigcirc^{\epsilon} Ep \wedge \ldots$$

(Fixed point of $C^{\epsilon}p \equiv \bigcirc^{\epsilon} EC^{\epsilon}p$)

  - attainable when there is a bound of ϵ on message delivery time.
- Eventual common knowledge:

$$C^{\diamond}p \equiv \diamond EC^{\diamond}p$$

  - Appropriate when there is no bound on message delivery time.
- Time stamped common knowledge

$$C^{TS}p \equiv E^{TS}C^{TS}p$$

  - For systems with clocks.

Can also have probabilistic variants and combinations.

Using eventual common knowledge

Theorem: Eventual common knowledge is not attainable in any system where communication is not guaranteed.

Back to coordinated attack ...

Corollary: Any protocol that guarantees that if one of the generals attacks, then eventually the other one will is necessarily a protocol where necessarily neither general attacks.

Fast forward 30 years . . .

The blockchain

At the heart of bitcoin is a *blockchain*, protocol for achieving consensus on a public ledger that records bitcoin transactions.

- Blockchain protocols can be used for applications like contract signing and for making transactions (like house sales) public.
- Contract signing is supposed to give agent *common knowledge*
  - Both signers know that both signers know . . . that the contract was signed
- Similarly, make a house sale public means make the sale common knowledge.

What is the semantics of a blockchain protocol?

- What properties do we want it to guarantee?
- **Claim:** these questions are best understood in terms of knowledge

Why it's subtle

A *ledger* is a distributed database that can be viewed as a sequence of blocks of data.

- Different agents typically have different views about which transactions are in the blockchain.
- With current blockchain protocols, it is also possible that a given transaction is included in agent i's view of the ledger at time m and not included at a later time m'.
- The set of agents involved changes over time.
- We need to allow for *dishonest* agents that do not follow the protocol, and may try to subvert it.
- We have *asynchrony*:
  - message delivery time is uncertain (although bounded)

We need to guarantee that a blockchain protocol gives us appropriate knowledge despite all this.

Typical assumptions

A ledger X is a T-*prefix* of a ledger Y if X is any prefix of the ledger that contains all but the last T transactions in Y.

Typical assumptions

A ledger X is a T-*prefix* of a ledger Y if X is any prefix of the ledger that contains all but the last T transactions in Y.

Blockchain protocols are assumed to be T-*consistent*:

- if i is honest (i.e., i has followed the protocol since joining the system) and X is a T-prefix of i's ledger at time m, then at all times $m' \geq m$, all honest agents will have X as a prefix of their ledger.

Does T-consistency suffice to use a blockchain protocol for the types of applications envisioned for it?

Typical assumptions

A ledger X is a T-*prefix* of a ledger Y if X is any prefix of the ledger that contains all but the last T transactions in Y.

Blockchain protocols are assumed to be T-*consistent*:

- if i is honest (i.e., i has followed the protocol since joining the system) and X is a T-prefix of i's ledger at time m, then at all times $m' \geq m$, all honest agents will have X as a prefix of their ledger.

Does T-consistency suffice to use a blockchain protocol for the types of applications envisioned for it?

- Spoiler alert: no!

Typical assumptions

A ledger X is a T-*prefix* of a ledger Y if X is any prefix of the ledger that contains all but the last T transactions in Y.

Blockchain protocols are assumed to be T-*consistent*:

- if i is honest (i.e., i has followed the protocol since joining the system) and X is a T-prefix of i's ledger at time m, then at all times $m' \geq m$, all honest agents will have X as a prefix of their ledger.

Does T-consistency suffice to use a blockchain protocol for the types of applications envisioned for it?

- Spoiler alert: no!

So what else do we need?

- That depends on what we want to achieve

A contract-signing example

- Suppose that attorneys require that electronic signatures on the contract are received by 11:30 AM on a global clock
- If they are received by then, the contract will be in force at noon on the global clock.

We might hope that if signatures are received by 11:30 AM, it is common knowledge that messages from the attorney are all received within at most 5 minutes, and everything is recorded on the ledger, then at noon on the global clock all agents will have common knowledge that the contract is in force.

A contract-signing example

- Suppose that attorneys require that electronic signatures on the contract are received by 11:30 AM on a global clock
- If they are received by then, the contract will be in force at noon on the global clock.

We might hope that if signatures are received by 11:30 AM, it is common knowledge that messages from the attorney are all received within at most 5 minutes, and everything is recorded on the ledger, then at noon on the global clock all agents will have common knowledge that the contract is in force.

Unfortunately, this does *not* follow from T-consistency:

- If $T = 10$ and the only transactions are the receipt of the messages and the contract being signed, it is compatible with T-consistency that the contract being signed is on one agent's ledger but never gets on the second agent's ledger.

Δ-weak growth

We need one more property to deal with this example:

- Δ-*weak growth* [Pass-Seeman-Shelat 2016]: if i is an honest agent and has a ledger of length N at time t, then all honest agents will have ledgers of length N by time $t + \Delta$.

Δ-weak growth

We need one more property to deal with this example:

- Δ-*weak growth* [Pass-Seeman-Shelat 2016]: if i is an honest agent and has a ledger of length N at time t, then all honest agents will have ledgers of length N by time $t + \Delta$.

Our main result: the combination of Δ-weak growth and T-consistency suffices not just for agent 1 to know that agent 2 will know (within time Δ) that 1 will have the contract in his ledger; the combination is necessary and sufficient to achieve Δ-$\Box$-*common knowledge* among the honest agents that the contract is in all of their ledgers.

- Roughly speaking, each honest agent knows that within Δ all the honest agents will know from that point on that within Δ all the honest agents will know from that point on $\ldots \phi$.
  - Even though the set of honest agents can change over time

This level of knowledge suffices to ensure coordination among honest agents within a window of Δ.

Back to runs and systems

When we defined the runs and systems framework earlier, we implicitly assumed that the set of agents is fixed and stable.

- That's not true in the blockchain world.

A more general framework:

- $\mathcal{AG}$ = all agents that could ever be in the system
- $\mathcal{A}(r, m)$ = the agents actually present in run r at time m.
- $\mathcal{H}(r, m) \subseteq \mathcal{A}(r, m)$ consists of the honest agents at (r, m)
  - $\mathcal{H}$ and $\mathcal{A}$ are *indexical sets*;
  - they can shrink or grow over time
- At (r, m), each agent in $\mathcal{A}(r, m)$ is in some *local state*
- The *global state* at (r, m) is $\{(s_i, i) : i \in \mathcal{A}(r, m)\}$
  - The set of local states of agents $i \in \mathcal{A}(r, m)$
- Let $r_i(m) = s_i$ (for $i \in A(r, m)$)

Interpreted systems

To reason about a blockchain protocol, we start with primitive propositions

- $i \in \mathcal{H}$: $(\mathcal{R}, r, m) \models i \in \mathcal{H}$ if $i \in \mathcal{H}(r, m)$
- $T\text{-}prefix(X, L_i)$: $(\mathcal{R}, r, m) \models T\text{-}prefix(X, L_i)$ if X is a T-prefix of $L_i(r, m)$, i's view of the ledger at time m in run r

Interpreted systems

To reason about a blockchain protocol, we start with primitive propositions

- $i \in \mathcal{H}$: $(\mathcal{R}, r, m) \models i \in \mathcal{H}$ if $i \in \mathcal{H}(r, m)$
- $T\text{-}prefix(X, L_i)$: $(\mathcal{R}, r, m) \models T\text{-}prefix(X, L_i)$ if X is a T-prefix of $L_i(r, m)$, i's view of the ledger at time m in run r

Non-epistemic operators:

- $(\mathcal{R}, r, m) \models \Box\phi$ iff $(\mathcal{R}, r, m') \models \phi$ for all $m' \geq m$
- $(\mathcal{R}, r, m) \models \bigcirc^{\Delta}\phi$ iff $(\mathcal{R}, r, m + \Delta) \models \phi$.

Interpreted systems

To reason about a blockchain protocol, we start with primitive propositions

- $i \in \mathcal{H}$: $(\mathcal{R}, r, m) \models i \in \mathcal{H}$ if $i \in \mathcal{H}(r, m)$
- $T\text{-}prefix(X, L_i)$: $(\mathcal{R}, r, m) \models T\text{-}prefix(X, L_i)$ if X is a T-prefix of $L_i(r, m)$, i's view of the ledger at time m in run r

Non-epistemic operators:

- $(\mathcal{R}, r, m) \models \Box\phi$ iff $(\mathcal{R}, r, m') \models \phi$ for all $m' \geq m$
- $(\mathcal{R}, r, m) \models \bigcirc^{\Delta}\phi$ iff $(\mathcal{R}, r, m + \Delta) \models \phi$.

Proposition: Protocol P is T-consistent and satisfies Δ-weak growth iff for all $i, j \in \mathcal{AG}$, the formula

$$i \in \mathcal{H} \wedge T\text{-}prefix(X, L_i) \Rightarrow \bigcirc^{\Delta}\Box(j \in \mathcal{H} \Rightarrow T\text{-}prefix(X, L_j))$$

is valid in $\mathcal{R}_P$.

- $\mathcal{R}_P$ is the system corresponding to protocol P

Epistemic operators

But what do agents *know* if they run a blockchain protocol?

Suppose that $\mathcal{S}$ is an indexical set:

- $(\mathcal{R}, r, m) \models B_i^{\mathcal{S}}\phi$ iff $(\mathcal{R}, r', m') \models \phi$ for all (r', m') such that $r_i(m) = r'_i(m)$ and $i \in \mathcal{S}(r', m')$.
  - i knows that if $i \in \mathcal{S}$, then ϕ holds
  - idea for definition due to Moses and Tuttle [1988]
- $E_{\mathcal{S}}\phi =_{\text{def}} \wedge_{i\in\mathcal{S}} B_i^{\mathcal{S}}\phi$
- $C_{\mathcal{S}}\phi =_{\text{def}} \wedge_{n=1}^{\infty} E_{\mathcal{S}}^n\phi$

More general notion:

- $C_{\mathcal{S}}^{\bigcirc^{\Delta}\Box}\phi =_{\text{def}} \wedge_{n=1}^{\infty}(\bigcirc^{\Delta}\Box E_{\mathcal{S}}\phi)^n$
  - Δ-$\Box$ common knowledge among the players in $\mathcal{S}$.

Towards an epistemic characterization

We want to prove that, for all i, j

$$i \in \mathcal{H} \wedge T\text{-}prefix(X, L_i) \Rightarrow C_{\mathcal{H}}^{\bigcirc^{\Delta}\Box}(j \in \mathcal{H} \Rightarrow T\text{-}prefix(X, L_j)).$$

- if i is honest then everything in i's T-prefix is Δ-$\Box$ common knowledge among the honest players
  - within Δ, all the honest players will know that from then on, within Δ, all the honest players will know ... everything in i's T-prefix

Towards an epistemic characterization

We want to prove that, for all i, j

$$i \in \mathcal{H} \wedge T\text{-}prefix(X, L_i) \Rightarrow C_{\mathcal{H}}^{\bigcirc^{\Delta}\Box}(j \in \mathcal{H} \Rightarrow T\text{-}prefix(X, L_j)).$$

- if i is honest then everything in i's T-prefix is Δ-$\Box$ common knowledge among the honest players
  - within Δ, all the honest players will know that from then on, within Δ, all the honest players will know ... everything in i's T-prefix

Standard way to prove common knowledge:
Lemma: $i \in \mathcal{H} \wedge \psi \Rightarrow \bigcirc^{\Delta}\Box E_{\mathcal{H}}\psi$ is valid for all $i \in \mathcal{H}$, then so is $i \in \mathcal{H} \wedge \psi \Rightarrow C_{\mathcal{H}}^{\bigcirc^{\Delta}\Box}\psi$.
Problem: What is ψ? $T\text{-}prefix(X, L_i)$? $T\text{-}prefix(X, L_j)$

- The formulas $T\text{-}prefix(X, L_j)$ are different for each j
- But they're similar!
  - They say "X is in 'my' T-prefix"
- If we change the language slightly, they become the same!

Agent-relative formulas

We allow *agent-relative* formulas

- Their truth depends on the agent

Have two new primitive propositions:

- $I \in \mathcal{H}$ ("I am honest")
  - $(\mathcal{R}, r, m, i) \models I \in \mathcal{H}$ if $i \in \mathcal{H}(r, m)$
- $T\text{-}prefix(X, L)$ ("X is in a T-prefix of my ledger")
  - $(\mathcal{R}, r, m, i) \models T\text{-}prefix(X, L)$ if X is a T-prefix of $L_i(r, m)$

Can prove the validity of

$$I \in \mathcal{H} \wedge T\text{-}prefix(X, L) \Rightarrow C^{\bigcirc^{\Delta}\Box}(T\text{-}prefix(X, L)).$$

This gives us the desired epistemic characterization of the blockchain protocol.

Adding probability

In practice, T-consistency and Δ-weak growth are not guaranteed to hold.

- They are only guaranteed to hold with high probability

Adding probability

In practice, T-consistency and Δ-weak growth are not guaranteed to hold.

- They are only guaranteed to hold with high probability

We can characterize the knowledge of agents using a blockchain protocol with probabilistic beliefs by considering probabilistic variants of common knowledge

- With high probability, within Δ everybody knows from then on that with high probability, within Δ ...

There are some subtleties in defining this in an asynchronous setting.

- See the full paper

Discussion

We got what we thought we wanted. Did we get what we needed?

Discussion

We got what we thought we wanted. Did we get what we needed?

Not necessarily:

- We may also want Δ'-*liveness*
  - If i wants to add something to a ledger, then within Δ' it is added
- May want to prevent ledgers from growing too quickly
  - So that the Nth transaction for i is close to the Nth transaction for j

But for many contract signing applications, Δ-$\Box$ common knowledge is just what we need.

Example: Suppose that two players want to sign a contract if either gets some signal (in their ledger).

- If both sign within some small interval Δ after at least one gets a signal, then they both get high utility.
- If one signs but the other doesn't sign soon enough, both get large negative utility.
- if one player signs before a signal is received or signs without the other player signing, then that player gets large negative utility.
- a player who doesn't sign gets utility 0.
- The signing is external to the ledger.

A player who gets a signal signs, and sends a message to the other player to sign, who signs when he gets the message.

- They are signing when Δ-$\Box$ common knowledge holds.