Verification of population protocols with unordered data

Steffen van Bergerem
Roland Guttenberg
Sandra Kiefer
Corto Mascle (thanks Corto for the slides!)
Nicolas Waldburger
Chana Weil-Kennedy

Published at ICALP'24

How to pass a law in the dark

MPs want to know whether a majority of them are in favour of a law proposal.

How to pass a law in the dark

MPs want to know whether a majority of them are in favour of a law proposal.

Four states: AY, PY, AN, PN: Active/Passive, Yes/No
Initially everyone is in the active state corresponding to their opinion.

How to pass a law in the dark

MPs want to know whether a majority of them are in favour of a law proposal.

Four states: AY, PY, AN, PN: Active/Passive, Yes/No
Initially everyone is in the active state corresponding to their opinion.
Interactions:

$$\begin{aligned}
AY + AN &\to PY + PN \\
AY + PN &\to AY + PY \\
AN + PY &\to AN + PN \\
PN + PY &\to PN + PN
\end{aligned}$$

How to pass a law in the dark

MPs want to know whether a majority of them are in favour of a law proposal.

Four states: AY, PY, AN, PN: Active/Passive, Yes/No
Initially everyone is in the active state corresponding to their opinion.
Interactions:

$$
\begin{aligned}
AY + AN &\to PY + PN \\
AY + PN &\to AY + PY \\
AN + PY &\to AN + PN \\
PN + PY &\to PN + PN
\end{aligned}
$$

AY AY

AN AN

AY

AN

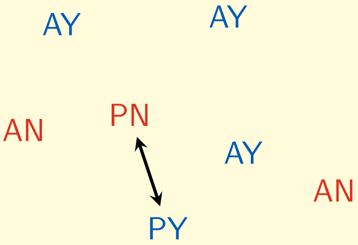
AY

How to pass a law in the dark

MPs want to know whether a majority of them are in favour of a law proposal.

Four states: AY, PY, AN, PN: Active/Passive, Yes/No
Initially everyone is in the active state corresponding to their opinion.
Interactions:

$$
\begin{aligned}
AY + AN &\to PY + PN \\
AY + PN &\to AY + PY \\
AN + PY &\to AN + PN \\
PN + PY &\to PN + PN
\end{aligned}
$$

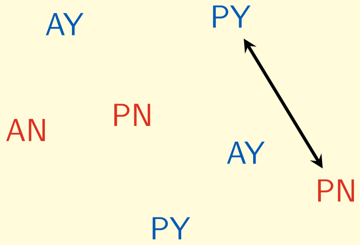


How to pass a law in the dark

MPs want to know whether a majority of them are in favour of a law proposal.

Four states: AY, PY, AN, PN: Active/Passive, Yes/No
Initially everyone is in the active state corresponding to their opinion.
Interactions:

$$AY + AN \rightarrow PY + PN$$
$$AY + PN \rightarrow AY + PY$$
$$AN + PY \rightarrow AN + PN$$
$$PN + PY \rightarrow PN + PN$$

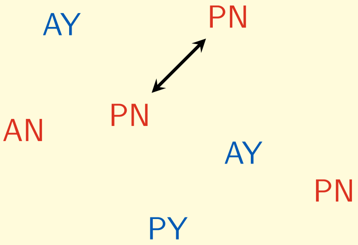


How to pass a law in the dark

MPs want to know whether a majority of them are in favour of a law proposal.

Four states: AY, PY, AN, PN: Active/Passive, Yes/No
Initially everyone is in the active state corresponding to their opinion.
Interactions:

$$
\begin{aligned}
AY + AN &\to PY + PN \\
AY + PN &\to AY + PY \\
AN + PY &\to AN + PN \\
PN + PY &\to PN + PN
\end{aligned}
$$

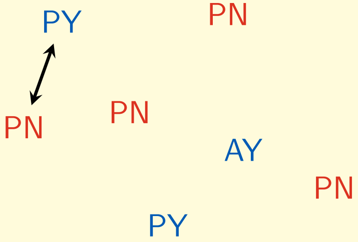


How to pass a law in the dark

MPs want to know whether a majority of them are in favour of a law proposal.

Four states: AY, PY, AN, PN: Active/Passive, Yes/No
Initially everyone is in the active state corresponding to their opinion.
Interactions:

$$AY + AN \to PY + PN$$
$$AY + PN \to AY + PY$$
$$AN + PY \to AN + PN$$
$$PN + PY \to PN + PN$$

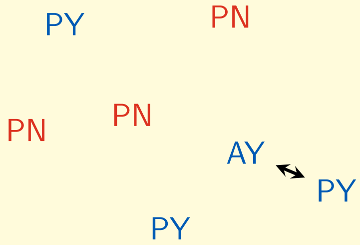


How to pass a law in the dark

MPs want to know whether a majority of them are in favour of a law proposal.

Four states: AY, PY, AN, PN: Active/Passive, Yes/No
Initially everyone is in the active state corresponding to their opinion.
Interactions:

$$
\begin{aligned}
AY + AN &\rightarrow PY + PN \\
AY + PN &\rightarrow AY + PY \\
AN + PY &\rightarrow AN + PN \\
PN + PY &\rightarrow PN + PN
\end{aligned}
$$

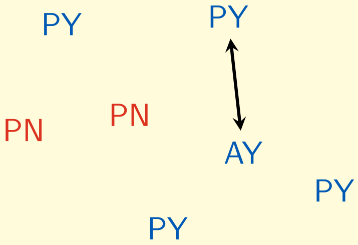


How to pass a law in the dark

MPs want to know whether a majority of them are in favour of a law proposal.

Four states: AY, PY, AN, PN: Active/Passive, Yes/No
Initially everyone is in the active state corresponding to their opinion.
Interactions:

$$
\begin{aligned}
AY + AN &\to PY + PN \\
AY + PN &\to AY + PY \\
AN + PY &\to AN + PN \\
PN + PY &\to PN + PN
\end{aligned}
$$

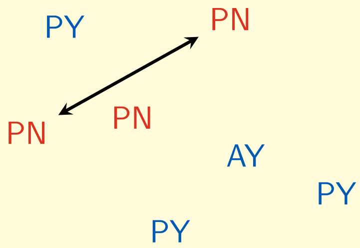


How to pass a law in the dark

MPs want to know whether a majority of them are in favour of a law proposal.

Four states: AY, PY, AN, PN: Active/Passive, Yes/No
Initially everyone is in the active state corresponding to their opinion.
Interactions:

$$
\begin{aligned}
AY + AN &\to PY + PN \\
AY + PN &\to AY + PY \\
AN + PY &\to AN + PN \\
PN + PY &\to PN + PN
\end{aligned}
$$

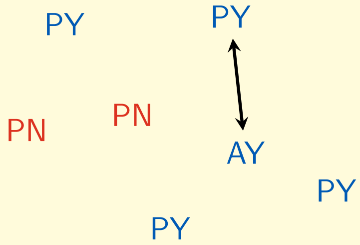


How to pass a law in the dark

MPs want to know whether a majority of them are in favour of a law proposal.

Four states: AY, PY, AN, PN: Active/Passive, Yes/No
Initially everyone is in the active state corresponding to their opinion.
Interactions:

$$
\begin{aligned}
AY + AN &\to PY + PN \\
AY + PN &\to AY + PY \\
AN + PY &\to AN + PN \\
PN + PY &\to PN + PN
\end{aligned}
$$

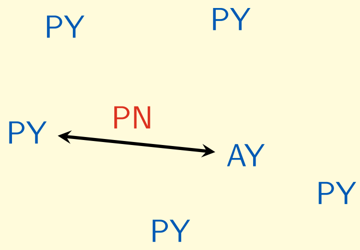


How to pass a law in the dark

MPs want to know whether a majority of them are in favour of a law proposal.

Four states: AY, PY, AN, PN: Active/Passive, Yes/No
Initially everyone is in the active state corresponding to their opinion.
Interactions:

$$AY + AN \to PY + PN$$
$$AY + PN \to AY + PY$$
$$AN + PY \to AN + PN$$
$$PN + PY \to PN + PN$$

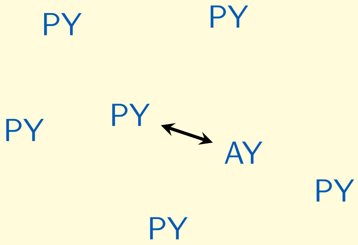


How to pass a law in the dark

MPs want to know whether a majority of them are in favour of a law proposal.

Four states: AY, PY, AN, PN: Active/Passive, Yes/No
Initially everyone is in the active state corresponding to their opinion.
Interactions:

$$
\begin{aligned}
AY + AN &\to PY + PN \\
AY + PN &\to AY + PY \\
AN + PY &\to AN + PN \\
PN + PY &\to PN + PN
\end{aligned}
$$



Population Protocols [Angluin, Aspnes, Diamadi, Fischer, Peralta, PODS 2004]

Finite set of states Q, with set $I \subseteq Q$ of *initial states*.
States are partitioned in two opinions $Q = Q_{Yes} \sqcup Q_{No}$
Interactions $\Delta \subseteq Q^2 \times Q^2$.

$$\left.\begin{matrix} q_0 \\ \\ q_2 \end{matrix}\right\} \longrightarrow \left\{\begin{matrix} q_1 \\ \\ q_3 \end{matrix}\right.$$

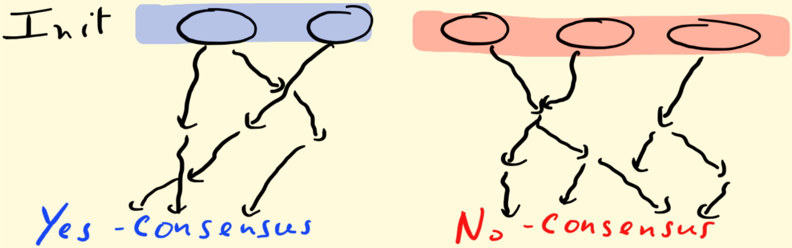
Population Protocols [Angluin, Aspnes, Diamadi, Fischer, Peralta, PODS 2004]

Finite set of states Q, with set $I \subseteq Q$ of *initial states*.
States are partitioned in two opinions $Q = Q_{Yes} \sqcup Q_{No}$
Interactions $\Delta \subseteq Q^2 \times Q^2$.

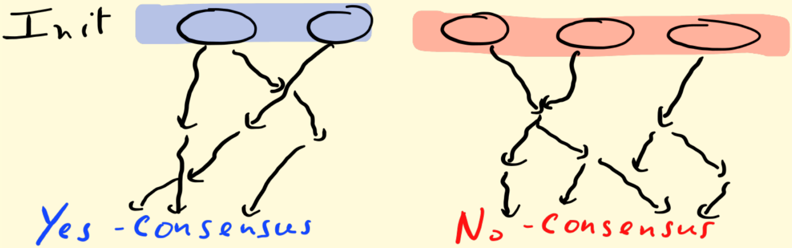
$$\left.\begin{matrix} q_0 \\ \\ q_2 \end{matrix}\right\} \longrightarrow \left\{\begin{matrix} q_1 \\ \\ q_3 \end{matrix}\right.$$

- Random pairwise interactions
- Stable consensus is reached when everyone agrees on Yes or No and no one can ever change their mind

A protocol is **well-specified** if from all initial configuration, either a Yes-consensus is reach with proba 1, or a No-consensus is reached with proba 1.



A protocol is **well-specified** if from all initial configuration, either a Yes-consensus is reach with proba 1, or a No-consensus is reached with proba 1.



The **predicate** computed by the protocol is then the set of initial configurations from which we reach a Yes-consensus.

First questions

Which predicates can be computed by population protocols?

First questions

Which predicates can be computed by population protocols? ✔

Theorem [Angluin, Aspnes, Eisenstat, Ruppert 2007]

A predicate is computable by a population protocol iff it is Presburger-definable.

First questions

Which predicates can be computed by population protocols? ✔

Theorem [Angluin, Aspnes, Eisenstat, Ruppert 2007]

A predicate is computable by a population protocol iff it is Presburger-definable.

Can we check if a population protocol is well-specified?

First questions

Which predicates can be computed by population protocols? ✔

Theorem [Angluin, Aspnes, Eisenstat, Ruppert 2007]

A predicate is computable by a population protocol iff it is Presburger-definable.

Can we check if a population protocol is well-specified? ✔

Theorem [Esparza, Ganty, Leroux, Majumdar 2015]

Checking if a population protocol is well-specified is **decidable** but as hard as Petri net reachability (Ackermann-complete).

Population Protocols with Unordered Data

Defined by Michael Blondin and François Ladouceur [ICALP'23]
Each agent carries a permanent datum taken from an infinite set $\mathbb{D}$.
Interactions: $\Delta \subseteq Q^2 \times \{=, \neq\} \to Q^2$
Interactions take into account whether the two agents have $=$ or $\neq$ data.

$$\left.\begin{matrix} q_0, x \\ \\ q_2, y \end{matrix}\right\} \xrightarrow[x \neq y]{} \left\{\begin{matrix} q_1, x \\ \\ q_3, y \end{matrix}\right.$$

Majority predicate

Does some datum have more agents than all other combined?

Majority predicate

Does some datum have more agents than all other combined?

Theorem [Blondin, Ladouceur ICALP'23]

There is a PPUD deciding the majority predicate.

Majority predicate

Does some datum have more agents than all other combined?

Theorem [Blondin, Ladouceur ICALP'23]

There is a PPUD deciding the majority predicate.

- Pair agents of distinct data until a candidate majority datum emerges
- Inform everyone whether they are part of the candidate datum or not
- Apply binary majority protocol

Majority predicate

Does some datum have more agents than all other combined?

Theorem [Blondin, Ladouceur ICALP'23]

There is a PPUD deciding the majority predicate.

- Pair agents of distinct data until a candidate majority datum emerges
- Inform everyone whether they are part of the candidate datum or not
- Apply binary majority protocol

Open problem

What are the predicates computed by PPUD?

Well-specification problem

Given a PPUD, is it well-specified?

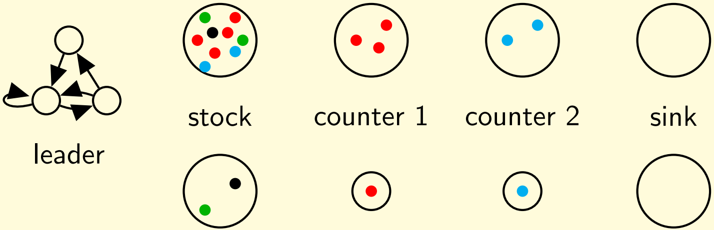
Well-specification problem

Given a PPUD, is it well-specified?

Theorem [Us, ICALP'24]

It is **undecidable** to check whether a PPUD is well-specified.

- Simulate a 2-counter machine with zero-tests.



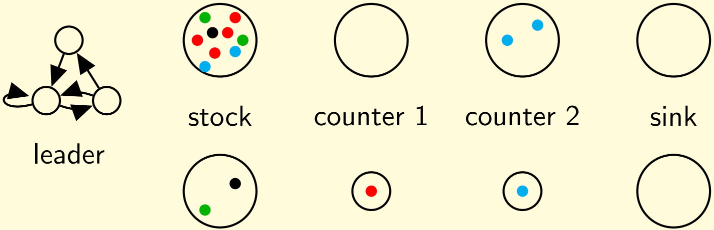
Well-specification problem

Given a PPUD, is it well-specified?

Theorem [Us, ICALP'24]

It is **undecidable** to check whether a PPUD is well-specified.

- Simulate a 2-counter machine with zero-tests.



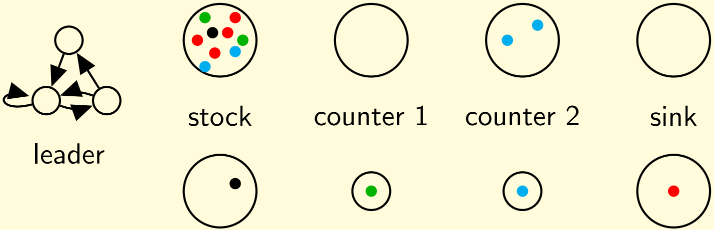
Well-specification problem

Given a PPUD, is it well-specified?

Theorem [Us, ICALP'24]

It is **undecidable** to check whether a PPUD is well-specified.

- Simulate a 2-counter machine with zero-tests.



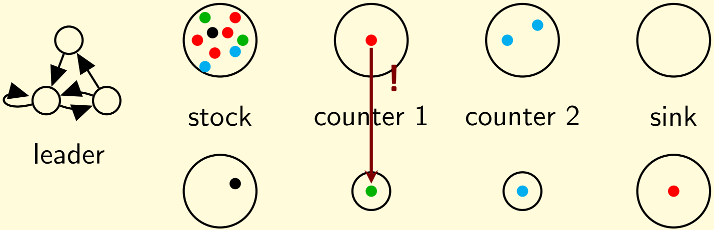
Well-specification problem

Given a PPUD, is it well-specified?

Theorem [Us, ICALP'24]

It is **undecidable** to check whether a PPUD is well-specified.

- Simulate a 2-counter machine with zero-tests.



Immediate Observation

A population protocol has the **Immediate Observation** property if in every interaction one of the two agents keeps the same state.

Immediate Observation

A population protocol has the **Immediate Observation** property if in every interaction one of the two agents keeps the same state.

$$\left.\begin{array}{l} q_0, x \\[2em] q_1, y \end{array}\right\} \xrightarrow[x \neq y]{} \left\{\begin{array}{l} q_0, x \\[2em] q_2, y \end{array}\right.$$

"observed agent"

Immediate Observation

A population protocol has the **Immediate Observation** property if in every interaction one of the two agents keeps the same state.

$$\left.\begin{array}{l} q_0, x \\ \\ q_1, y \end{array}\right\} \xrightarrow[x \neq y]{} \left\{\begin{array}{l} q_0, x \\ \\ q_2, y \end{array}\right. \qquad \text{“observed agent”}$$

Theorem (Esparza, Ganty, Majumdar, Weil-Kennedy 2018)

Well-specification is PSPACE-complete for Immediate-Observation population protocols **without data**.

Interval predicate = Boolean combination of
"At least 3 distinct data with between 1 and 3 agents in state q and 4 agents in state q'".

$$\exists d_1, d_2, d_3, \bigwedge_{i=1}^{3} (1 \leq \#(q, d_i) \leq 3) \wedge (4 \leq \#(q, d_i))$$

Interval predicate = Boolean combination of
"At least 3 distinct data with between 1 and 3 agents in state q and 4 agents in state q'".

$$\exists d_1, d_2, d_3, \bigwedge_{i=1}^{3} (1 \leq \#(q, d_i) \leq 3) \wedge (4 \leq \#(q, d_i))$$

Theorem [Blondin, Ladouceur 2023]

The predicates computed by IOPPUD are exactly interval predicates.

IOPPUD

Theorem [Us, ICALP'24]

Well-specification is decidable for IOPPUD.

IOPPUD

Theorem [Us, ICALP'24]

Well-specification is decidable for IOPPUD.

Key lemma

Given a set of configurations C described by an interval predicate, we can compute interval predicates expressing $Pre^*(C)$ and $Post^*(C)$.

IOPPUD

Theorem [Us, ICALP'24]

Well-specification is decidable for IOPPUD.

Key lemma

Given a set of configurations C described by an interval predicate, we can compute interval predicates expressing $Pre^*(C)$ and $Post^*(C)$.

Copycat: in an IOPPUD, if an agent with datum d goes from q_1 to q_2 then we can send as many agents with datum d as we want from q_1 to q_2: the observed agent is still here.

IOPPUD

Theorem [Us, ICALP'24]

Well-specification is decidable for IOPPUD.

Key lemma

Given a set of configurations C described by an interval predicate, we can compute interval predicates expressing $Pre^*(C)$ and $Post^*(C)$.

Copycat: in an IOPPUD, if an agent with datum d goes from q_1 to q_2 then we can send as many agents with datum d as we want from q_1 to q_2: the observed agent is still here.
Using this fact, we prove that we can rearrange any run so that

- each datum only has a limited number of agents that get observed during the run.
- only a limited number of data have agents that are observed by other data.

IOPPUD

Generalised Reachability Expressions:

$$E ::= \text{Interval Predicate} \mid E \cup E \mid \overline{E} \mid \mathit{Pre}^*(E) \mid \mathit{Post}^*(E)$$

Question: given a GRE E, do we have $[\![E]\!]_{\mathcal{P}}$?

IOPPUD

Generalised Reachability Expressions:

$$E ::= \text{Interval Predicate} \mid E \cup E \mid \overline{E} \mid \mathit{Pre}^*(E) \mid \mathit{Post}^*(E)$$

Question: given a GRE E, do we have $[\![E]\!]_{\mathcal{P}}$?

Example: The protocol is well-specified if and only if

$$\Gamma_0 \cap \mathit{Pre}^*(\overline{\mathit{Pre}^*(\text{Stable}_{\mathit{Yes}})}) \cap \mathit{Pre}^*(\overline{\mathit{Pre}^*(\text{Stable}_{\mathit{No}})}) = \emptyset$$

$\text{Stable}_b := \overline{\mathit{Pre}^*(\overline{\mathrm{Consensus}_b})}$: stable consensus on opinion b.

IOPPUD

Generalised Reachability Expressions:

$$E ::= \text{Interval Predicate} \mid E \cup E \mid \overline{E} \mid Pre^*(E) \mid Post^*(E)$$

Question: given a GRE E, do we have $[\![E]\!]_{\mathcal{P}}$?
Example: The protocol is well-specified if and only if

$$\Gamma_0 \cap Pre^*(\overline{Pre^*(\text{Stable}_{Yes})}) \cap Pre^*(\overline{Pre^*(\text{Stable}_{No})}) = \emptyset$$

$\text{Stable}_b := \overline{Pre^*(\overline{\mathrm{Consensus}_b})}$: stable consensus on opinion b.

Theorem

Given a GRE E, we can compute an interval predicate for $[\![E]\!]_{\mathcal{P}}$.

Corollary

Given a GRE E, we can check if $[\![E]\!]_{\mathcal{P}} = \emptyset$.

Decidable problems on IOPPUDs

Many problems can be reduced to the emptiness of a GRE:

- `Well-specification`
  = The protocol computes something

Decidable problems on IOPPUDs

Many problems can be reduced to the emptiness of a GRE:

- `Well-specification`

  $=$ The protocol computes something
- `Correctness`

  $=$ The protocol computes predicate P

Decidable problems on IOPPUDs

Many problems can be reduced to the emptiness of a GRE:

- `Well-specification`

  = The protocol computes something
- `Correctness`

  = The protocol computes predicate P
- `Visible termination`

  = all consensus are stable consensus

Decidable problems on IOPPUDs

Many problems can be reduced to the emptiness of a GRE:

- `Well-specification`
  = The protocol computes something
- `Correctness`
  = The protocol computes predicate P
- `Visible termination`
  = all consensus are stable consensus
- `Home-space problem`
  = Every fair run eventually reaches set of configurations H

Complexity

Emptiness of Generalised Reachability Expressions is:

In EXPSPACE

$\rightarrow$ By controlling the growth of coefficients when translating GRE to Interval Predicates.

NEXPTIME-hard

$\rightarrow$ By encoding the tiling of an exponential grid.

Open problems

- Characterise predicates computed by PPUD

Open problems

- Characterise predicates computed by PPUD
- Close the complexity gap for GRE emptiness in IOPPUD: NEXPTIME - EXPSPACE

Open problems

- Characterise predicates computed by PPUD
- Close the complexity gap for GRE emptiness in IOPPUD: NEXPTIME - EXPSPACE
- Close the complexity gap for well-specification in IOPPUD: PSPACE - EXPSPACE

Open problems

- Characterise predicates computed by PPUD
- Close the complexity gap for GRE emptiness in IOPPUD: NEXPTIME - EXPSPACE
- Close the complexity gap for well-specification in IOPPUD: PSPACE - EXPSPACE

Thanks!