Formal analysis of crowd systems

Michael Blondin

Université de Sherbrooke

Formal analysis of crowd systems

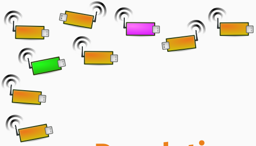
Michael Blondin

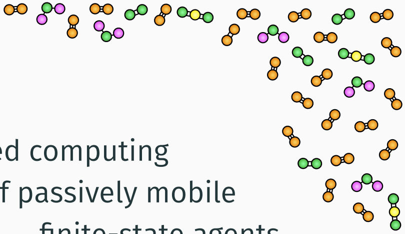
Joint work with J. Esparza, M. Helfrich, S. Jaax, A. Kučera, P. J. Meyer

Université de Sherbrooke

Population protocols: distributed computing model for massive networks of passively mobile finite-state agents

Overview

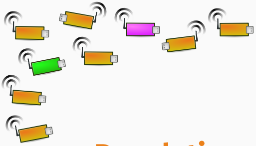


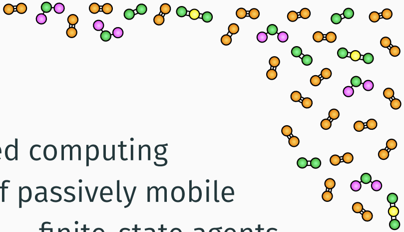


Population protocols: distributed computing model for massive networks of passively mobile finite-state agents

Model *e.g.* networks of passively mobile sensors and chemical reaction networks

Overview





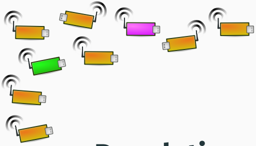
Population protocols: distributed computing model for massive networks of passively mobile finite-state agents

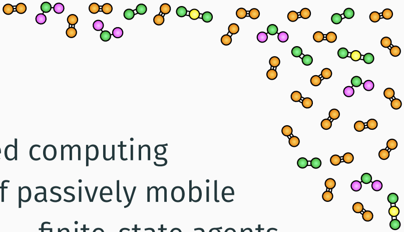
Model *e.g.* networks of passively mobile sensors and chemical reaction networks

Protocols compute predicates of the form $\varphi \colon \mathbb{N}^d \to \{0, 1\}$

e.g. $\varphi(m, n)$ is computed by $m + n$ agents

Overview





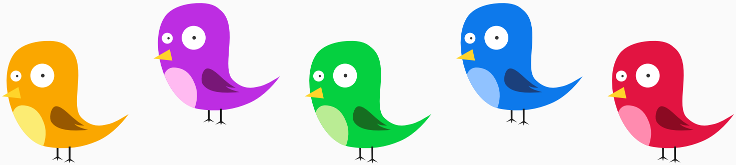
Population protocols: distributed computing model for massive networks of passively mobile finite-state agents

This talk: automatic verification and expected termination time analysis

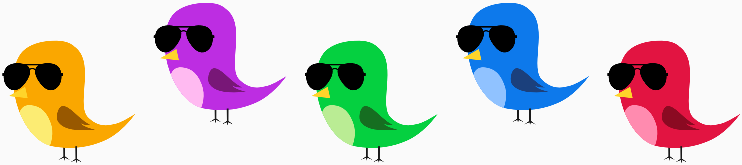
Population protocols

- anonymous mobile agents with very few resources
- agents change states via random pairwise interactions
- each agent has opinion true/false
- computes by stabilizing agents to some opinion

- anonymous mobile agents with very few resources
- agents change states via random pairwise interactions
- each agent has opinion true/false
- computes by stabilizing agents to some opinion

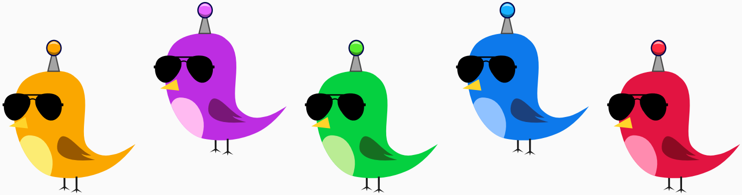


- anonymous mobile agents with very few resources
- agents change states via random pairwise interactions
- each agent has opinion true/false
- computes by stabilizing agents to some opinion



- anonymous mobile agents with very few resources
- agents change states via random pairwise interactions
- each agent has opinion true/false
- computes by stabilizing agents to some opinion

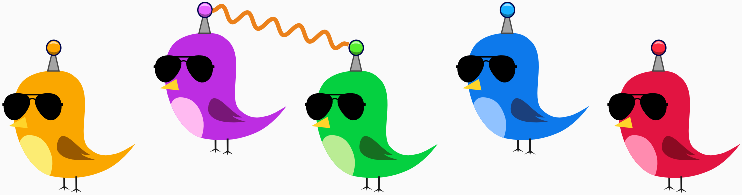
- anonymous mobile agents with very few resources
- agents change states via random pairwise interactions
- each agent has opinion true/false
- computes by stabilizing agents to some opinion



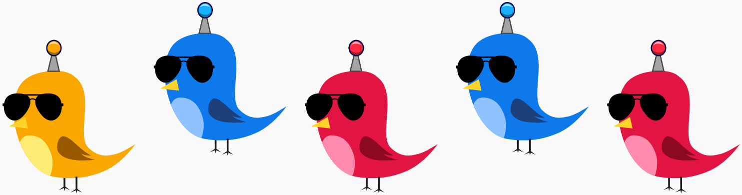
Population protocols

Angluin *et al.* PODC'04

- anonymous mobile agents with very few resources
- agents change states via random pairwise interactions
- each agent has opinion true/false
- computes by stabilizing agents to some opinion



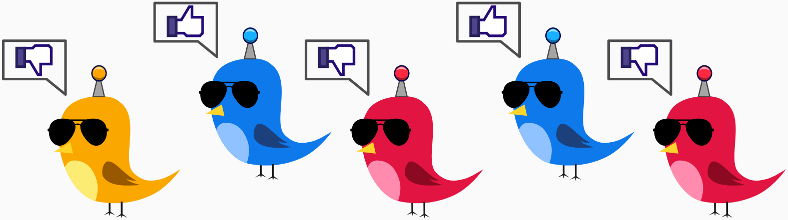
- anonymous mobile agents with very few resources
- agents change states via random pairwise interactions
- each agent has opinion true/false
- computes by stabilizing agents to some opinion



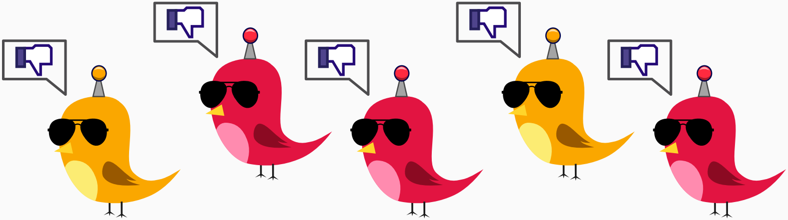
Population protocols

Angluin *et al.* PODC'04

- anonymous mobile agents with very few resources
- agents change states via random pairwise interactions
- each agent has opinion true/false
- computes by stabilizing agents to some opinion

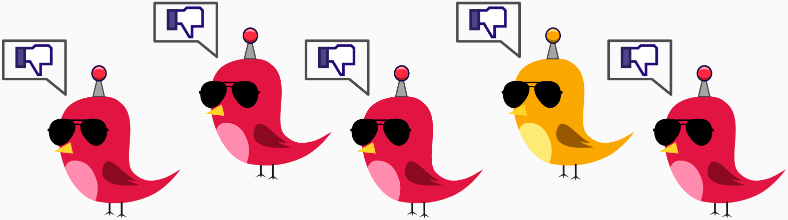


- anonymous mobile agents with very few resources
- agents change states via random pairwise interactions
- each agent has opinion true/false
- computes by stabilizing agents to some opinion



Population protocols

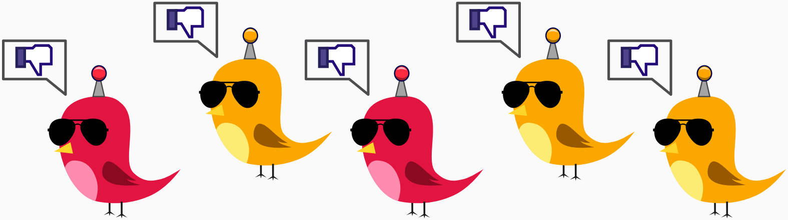
- anonymous mobile agents with very few resources
- agents change states via random pairwise interactions
- each agent has opinion true/false
- computes by stabilizing agents to some opinion



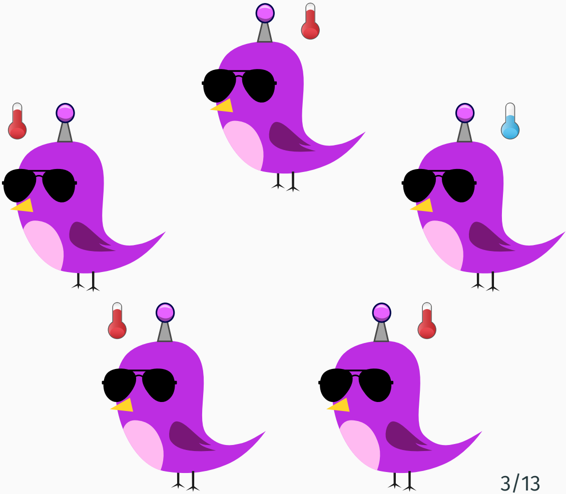
Population protocols

Angluin *et al.* PODC'04

- anonymous mobile agents with very few resources
- agents change states via random pairwise interactions
- each agent has opinion true/false
- computes by stabilizing agents to some opinion



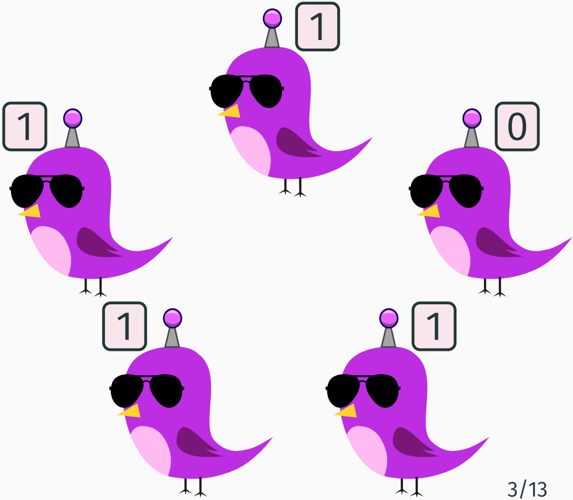
Are there at least 4 sick birds?



Are there at least 4 sick birds?

Protocol:

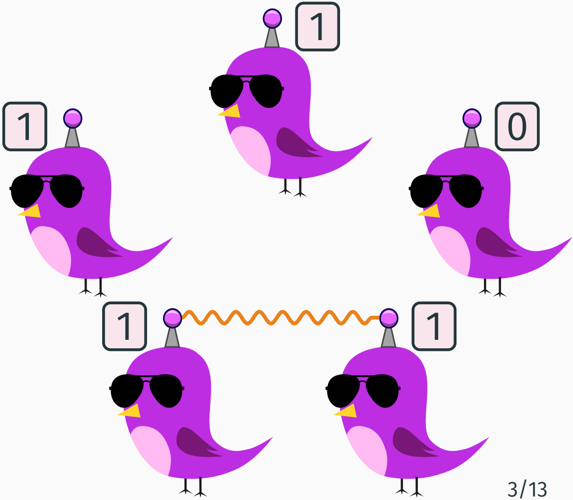
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$
  if $m + n < 4$
- $(m, n) \mapsto (4, 4)$
  if $m + n \geq 4$



Are there at least 4 sick birds?

Protocol:

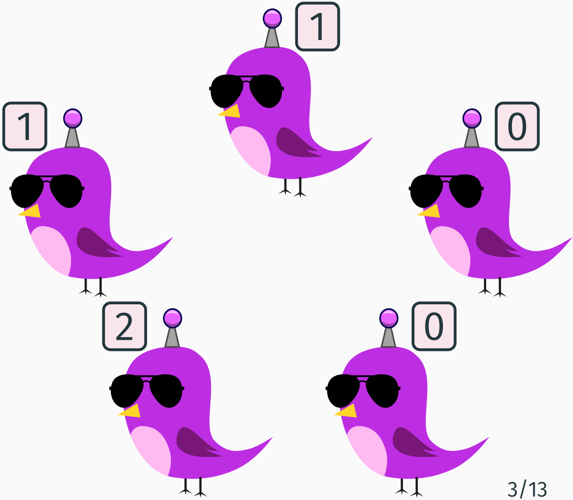
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$
  if $m + n < 4$
- $(m, n) \mapsto (4, 4)$
  if $m + n \geq 4$



Are there at least 4 sick birds?

Protocol:

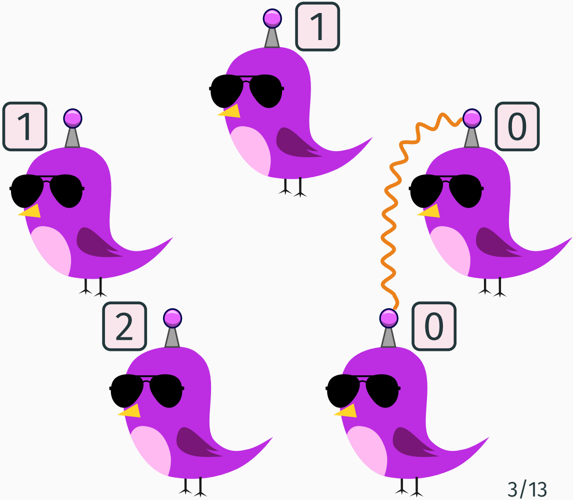
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$
  if $m + n < 4$
- $(m, n) \mapsto (4, 4)$
  if $m + n \geq 4$



Are there at least 4 sick birds?

Protocol:

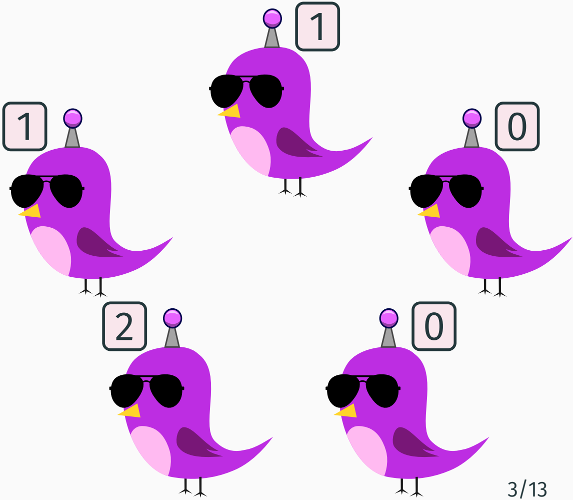
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$
  if $m + n < 4$
- $(m, n) \mapsto (4, 4)$
  if $m + n \geq 4$



Are there at least 4 sick birds?

Protocol:

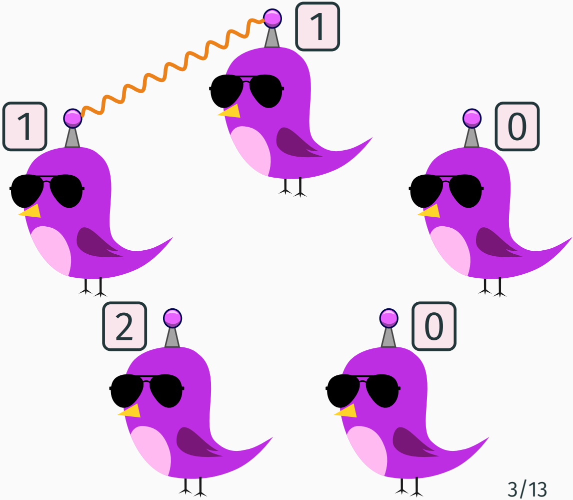
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$ if $m + n < 4$
- $(m, n) \mapsto (4, 4)$ if $m + n \geq 4$



Are there at least 4 sick birds?

Protocol:

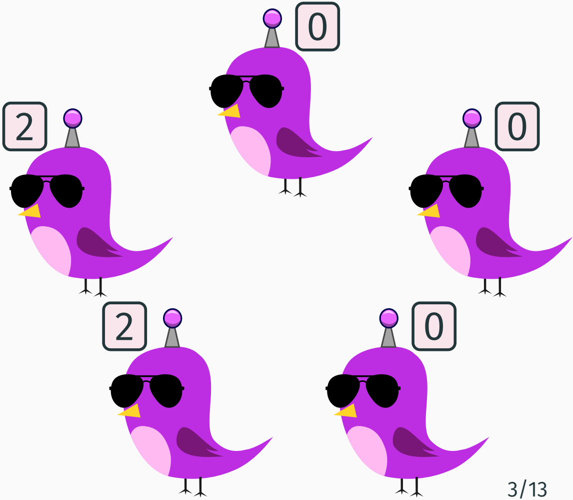
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$
  if $m + n < 4$
- $(m, n) \mapsto (4, 4)$
  if $m + n \geq 4$



Are there at least 4 sick birds?

Protocol:

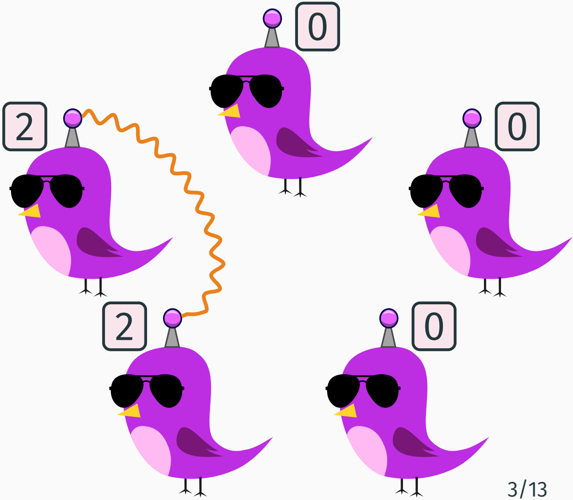
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$
  if $m + n < 4$
- $(m, n) \mapsto (4, 4)$
  if $m + n \geq 4$



Are there at least 4 sick birds?

Protocol:

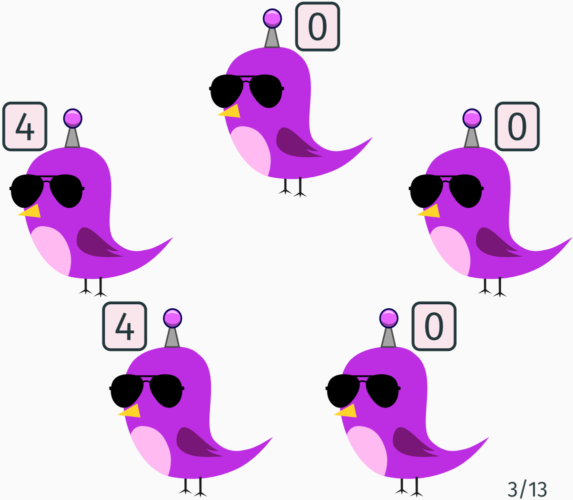
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$
  if $m + n < 4$
- $(m, n) \mapsto (4, 4)$
  if $m + n \geq 4$



Are there at least 4 sick birds?

Protocol:

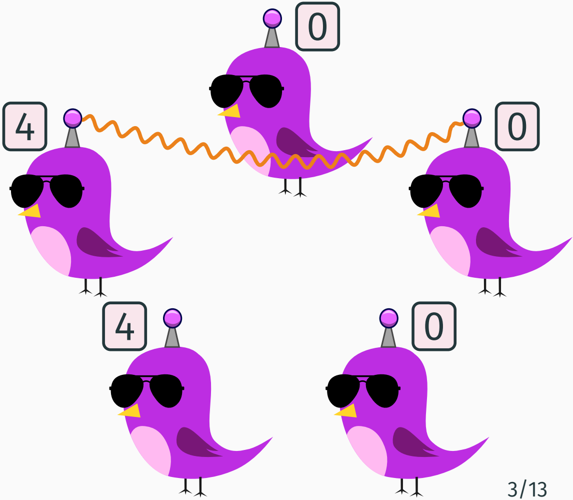
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$
  if $m + n < 4$
- $(m, n) \mapsto (4, 4)$
  if $m + n \geq 4$



Are there at least 4 sick birds?

Protocol:

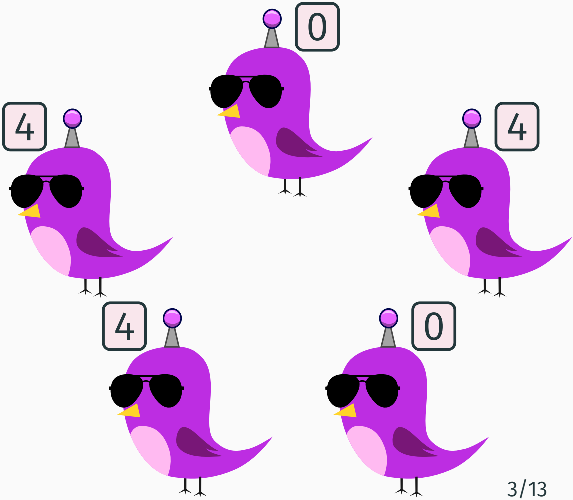
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$ if $m + n < 4$
- $(m, n) \mapsto (4, 4)$ if $m + n \geq 4$



Are there at least 4 sick birds?

Protocol:

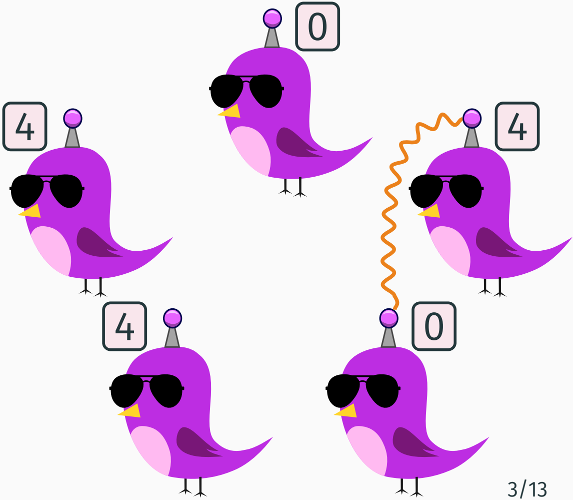
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$
  if $m + n < 4$
- $(m, n) \mapsto (4, 4)$
  if $m + n \geq 4$



Are there at least 4 sick birds?

Protocol:

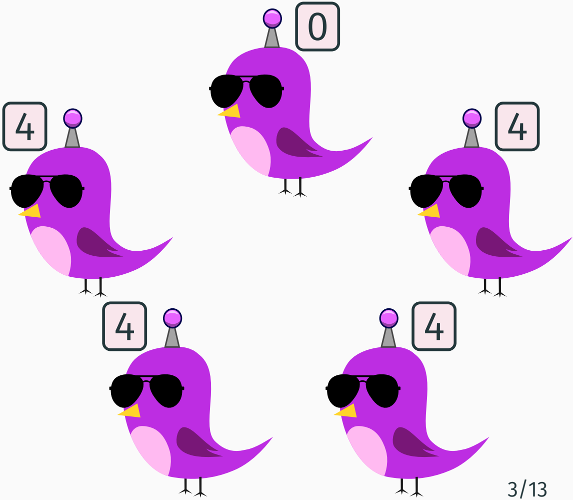
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$
  if $m + n < 4$
- $(m, n) \mapsto (4, 4)$
  if $m + n \geq 4$



Are there at least 4 sick birds?

Protocol:

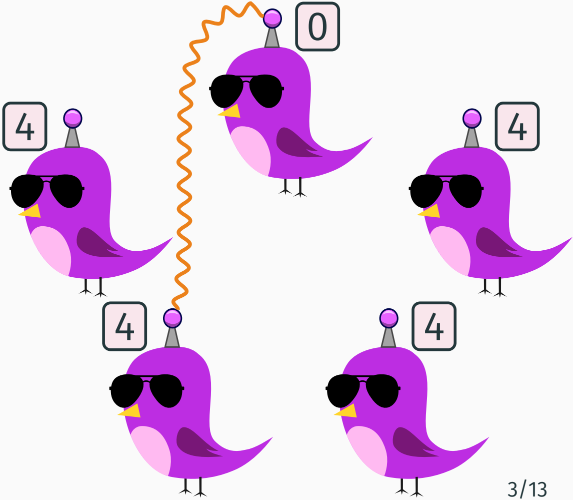
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$ if $m + n < 4$
- $(m, n) \mapsto (4, 4)$ if $m + n \geq 4$



Are there at least 4 sick birds?

Protocol:

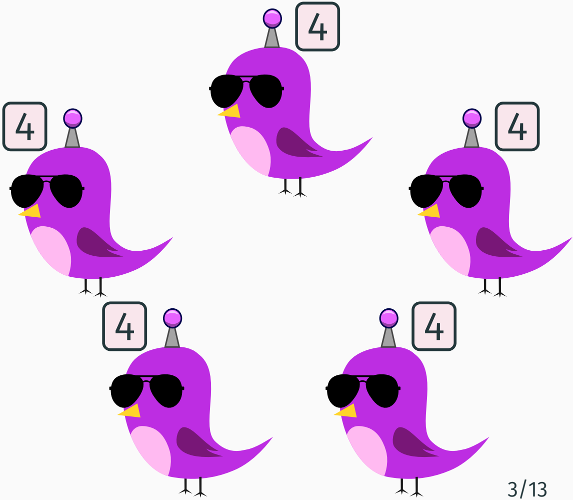
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$ if $m + n < 4$
- $(m, n) \mapsto (4, 4)$ if $m + n \geq 4$



Are there at least 4 sick birds?

Protocol:

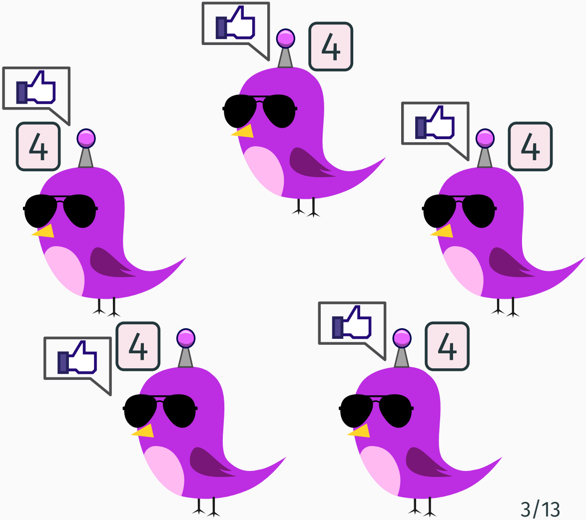
- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$
  if $m + n < 4$
- $(m, n) \mapsto (4, 4)$
  if $m + n \geq 4$



Are there at least 4 sick birds?

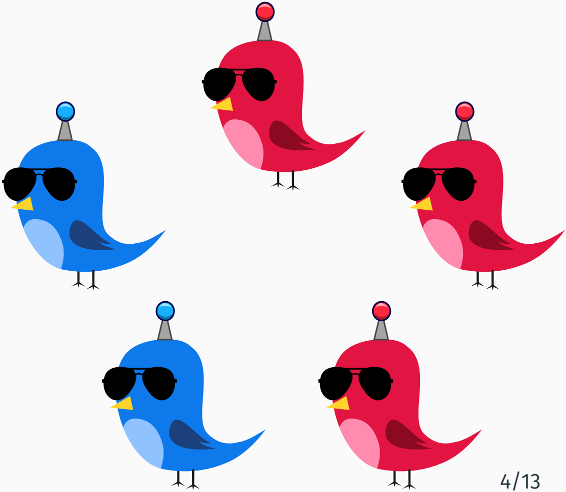
Protocol:

- Each agent in a state of $\{0, 1, 2, 3, 4\}$
- $(m, n) \mapsto (m + n, 0)$
  if $m + n < 4$
- $(m, n) \mapsto (4, 4)$
  if $m + n \geq 4$



Example: majority protocol

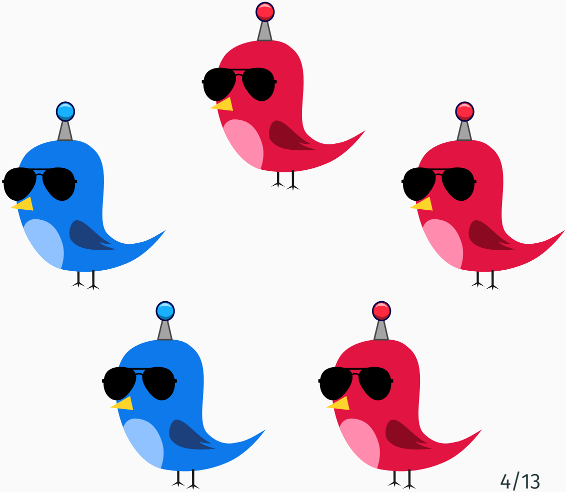
blue agents ≥ # red agents?



blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents
- Large agents convert small agents to their colour

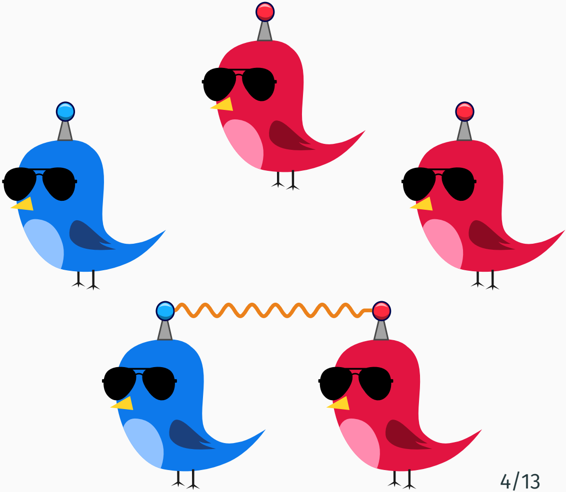


Example: majority protocol

blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents
- Large agents convert small agents to their colour

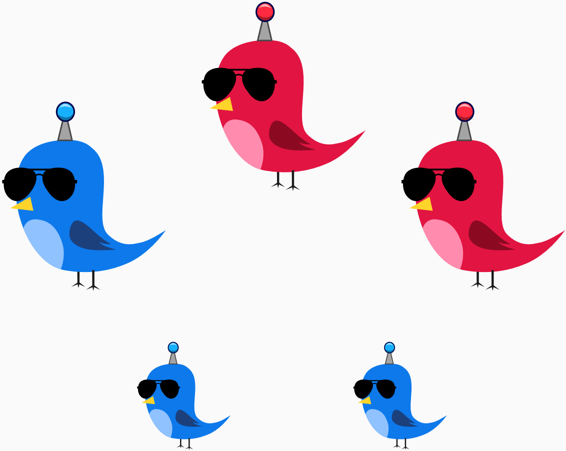


blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents

- Large agents convert small agents to their colour

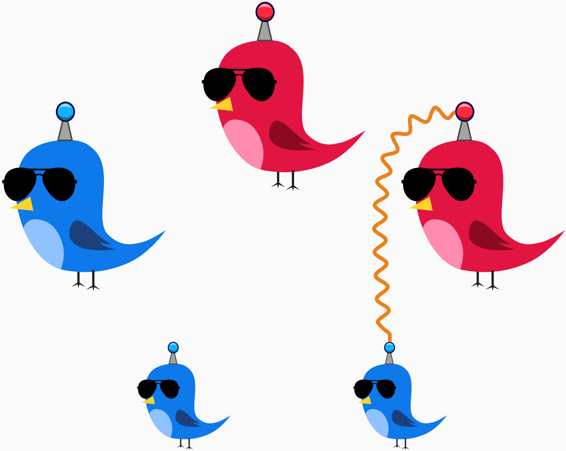


blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents

- Large agents convert small agents to their colour

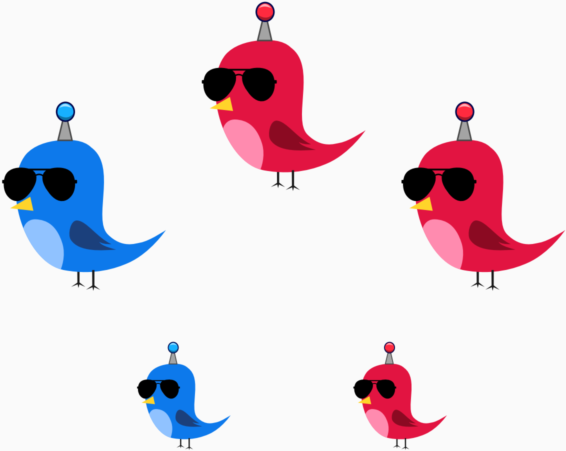


blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents

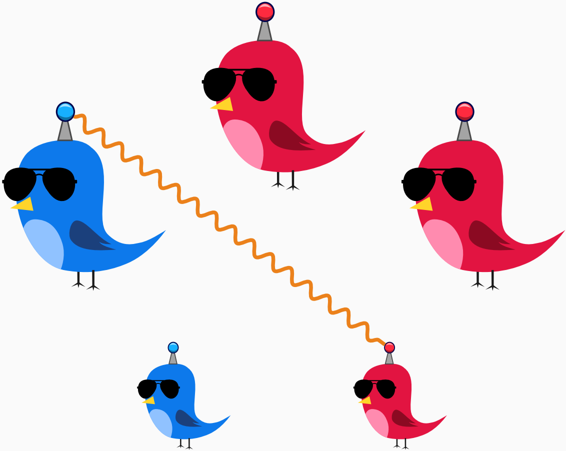
- Large agents convert small agents to their colour



blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents
- Large agents convert small agents to their colour

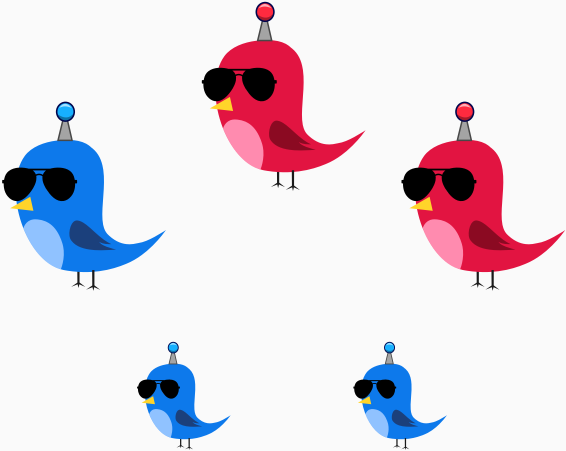


Example: majority protocol

blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents
- Large agents convert small agents to their colour

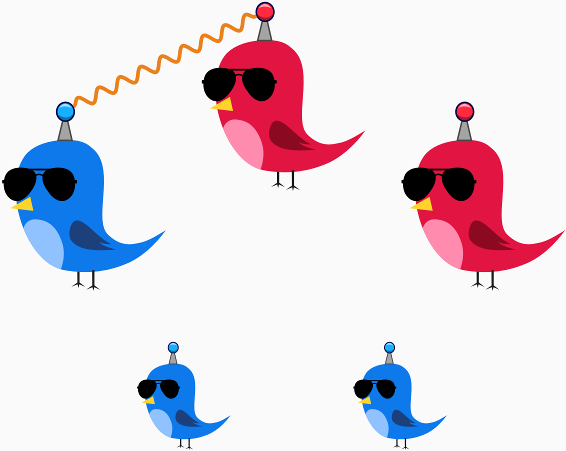


blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents

- Large agents convert small agents to their colour

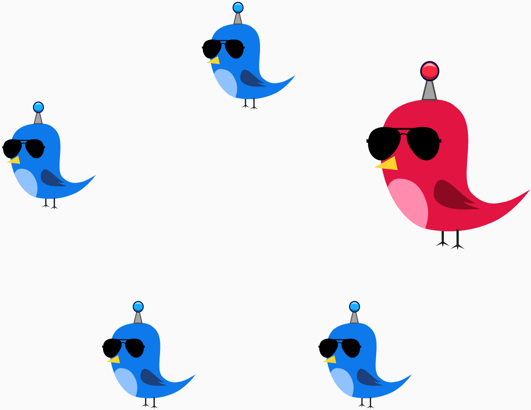


blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents

- Large agents convert small agents to their colour

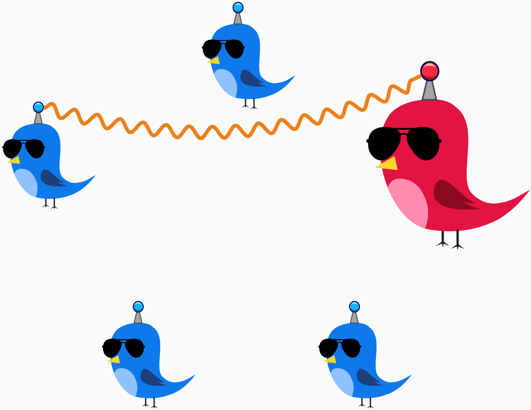


blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents

- Large agents convert small agents to their colour

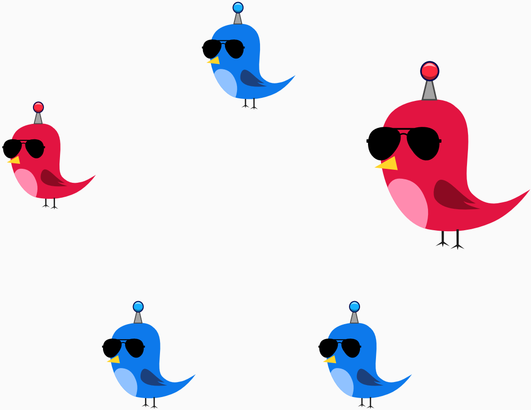


Example: majority protocol

blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents
- Large agents convert small agents to their colour



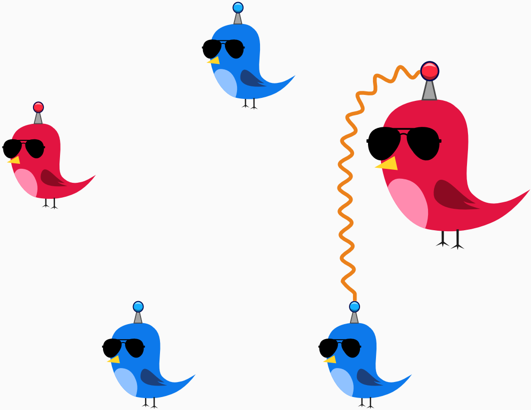
Example: majority protocol

blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents

- Large agents convert small agents to their colour

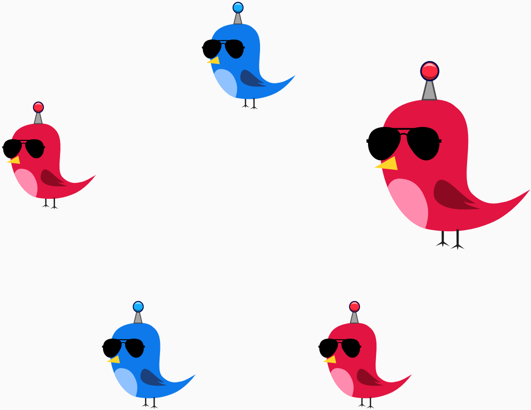


blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents

- Large agents convert small agents to their colour

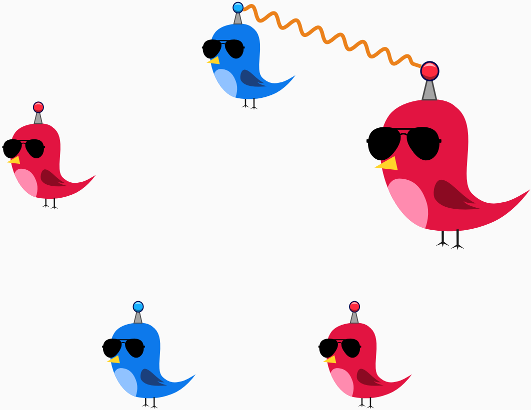


blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents

- Large agents convert small agents to their colour

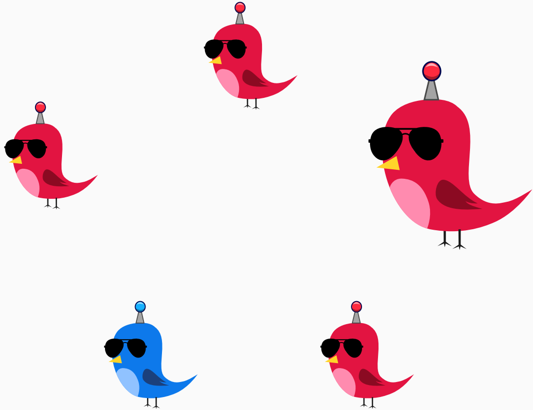


Example: majority protocol

blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents
- Large agents convert small agents to their colour

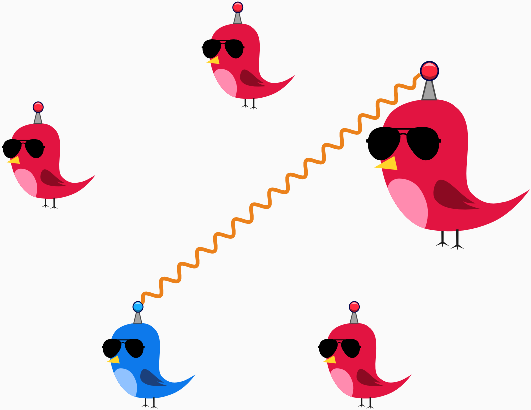


Example: majority protocol

blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents
- Large agents convert small agents to their colour

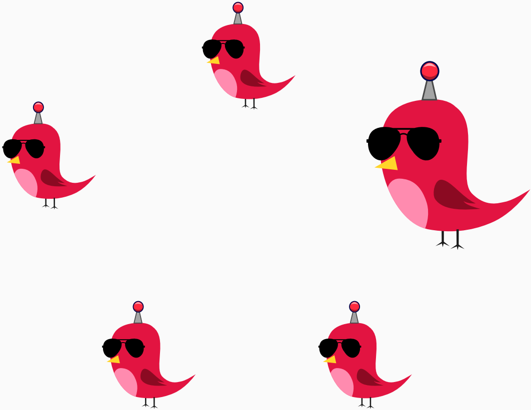


Example: majority protocol

blue agents ≥ # red agents?

Protocol:

- Two large agents become small blue agents
- Large agents convert small agents to their colour

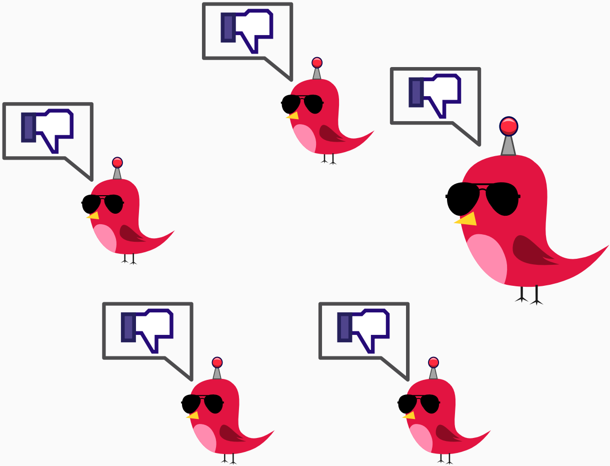


Example: majority protocol

blue agents ≥ # red agents?

Protocol:

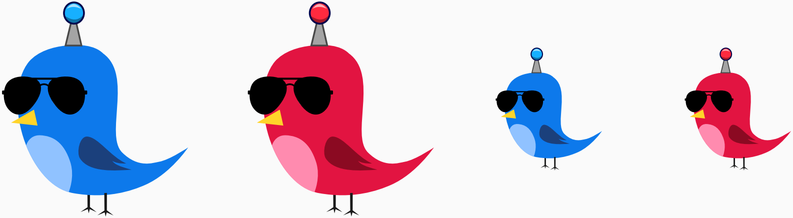
- Two large agents become small blue agents
- Large agents convert small agents to their colour



Demonstration

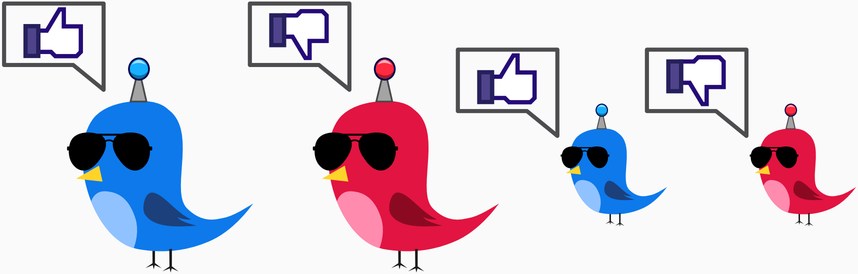
Population protocols: formal model

- *States*: finite set Q
- *Opinions*: $O\colon Q \to \{\mathtt{false}, \mathtt{true}\}$
- *Initial states*: $I \subseteq Q$
- *Transitions*: $T \subseteq Q^2 \times Q^2$



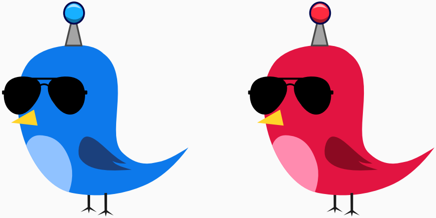
Population protocols: formal model

- *States*: finite set Q
- *Opinions*: $O\colon Q \to \{\texttt{false}, \texttt{true}\}$
- *Initial states*: $I \subseteq Q$
- *Transitions*: $T \subseteq Q^2 \times Q^2$



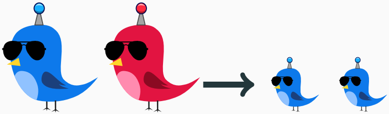
Population protocols: formal model

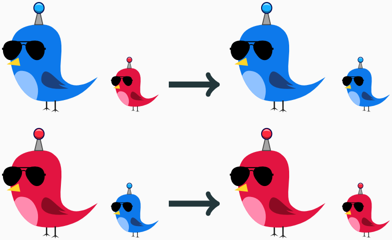
- *States*: finite set Q
- *Opinions*: $O\colon Q \to \{\texttt{false}, \texttt{true}\}$
- *Initial states*: $I \subseteq Q$
- *Transitions*: $T \subseteq Q^2 \times Q^2$



Population protocols: formal model

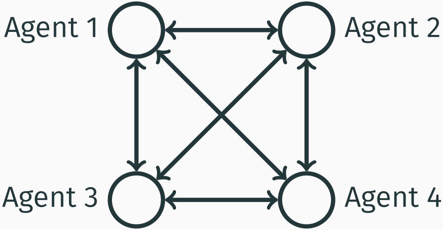
- *States*: finite set Q
- *Opinions*: $O\colon Q \to \{\mathtt{false}, \mathtt{true}\}$
- *Initial states*: $I \subseteq Q$
- *Transitions*: $T \subseteq Q^2 \times Q^2$





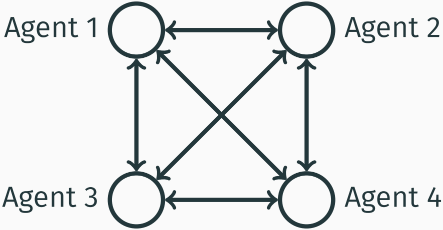
All agents can interact pairwise

(complete topology)



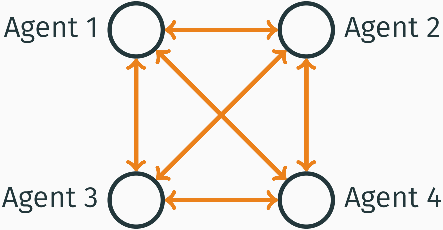
Population protocols: interactions

$$\mathbb{P}[\text{fire } p, q \mapsto p', q' \text{ in } C] = \begin{cases} \dfrac{2 \cdot C(p) \cdot C(q)}{n^2 - n} & \text{if } p \neq q \\[2ex] \dfrac{C(p) \cdot (C(p) - 1)}{n^2 - n} & \text{if } p = q \end{cases}$$



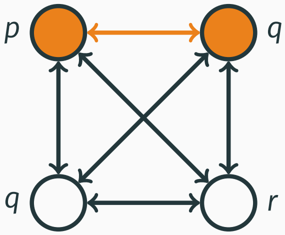
Population protocols: interactions

$$\mathbb{P}[\text{fire } p, q \mapsto p', q' \text{ in } C] = \begin{cases} \dfrac{2 \cdot C(p) \cdot C(q)}{n^2 - n} & \text{if } p \neq q \\[2ex] \dfrac{C(p) \cdot (C(p) - 1)}{n^2 - n} & \text{if } p = q \end{cases}$$



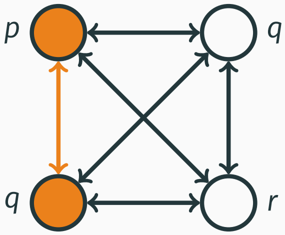
Population protocols: interactions

$$\mathbb{P}[\text{fire } p, q \mapsto p', q' \text{ in } C] = \begin{cases} \dfrac{2 \cdot C(p) \cdot C(q)}{n^2 - n} & \text{if } p \neq q \\[2ex] \dfrac{C(p) \cdot (C(p) - 1)}{n^2 - n} & \text{if } p = q \end{cases}$$



Population protocols: interactions

$$\mathbb{P}[\text{fire } p, q \mapsto p', q' \text{ in } C] = \begin{cases} \dfrac{2 \cdot C(p) \cdot C(q)}{n^2 - n} & \text{if } p \neq q \\[2ex] \dfrac{C(p) \cdot (C(p) - 1)}{n^2 - n} & \text{if } p = q \end{cases}$$

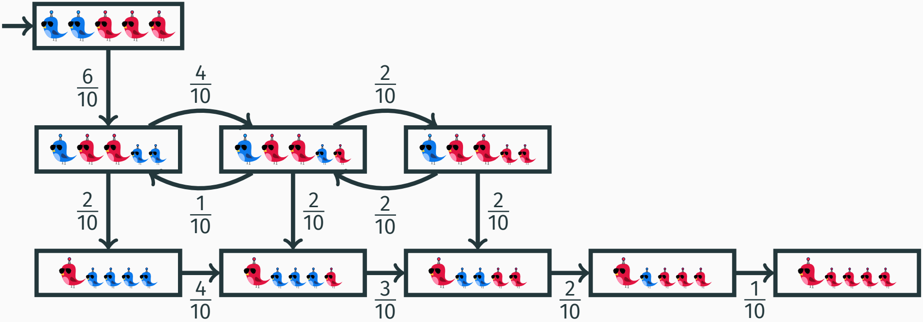


Population protocols: interactions

$$\mathbb{P}[\text{fire } p, q \mapsto p', q' \text{ in } C] \;=\; \begin{cases} \dfrac{2 \cdot C(p) \cdot C(q)}{n^2 - n} & \text{if } p \neq q \\[2ex] \dfrac{C(p) \cdot (C(p) - 1)}{n^2 - n} & \text{if } p = q \end{cases}$$

$$\mathbb{P}[C \to C'] \;=\; \sum_{t \text{ s.t. } C \xrightarrow{t} C'} \mathbb{P}[\text{fire } t \text{ in } C]$$

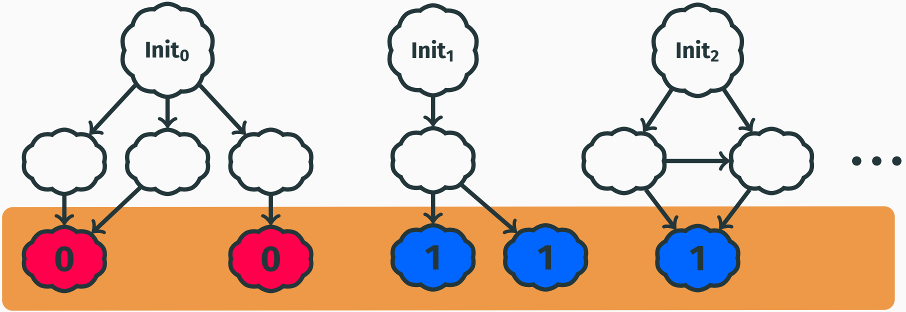
Underlying Markov chain:



A protocol computes a predicate $f\colon \mathbb{N}^I \to \{0,1\}$

if runs reach **common stable consensus**

with probability 1



A protocol computes a predicate $f\colon \mathbb{N}^I \to \{0,1\}$

if runs reach **common stable consensus**

with probability 1

Expressive power **Angluin, Aspnes, Eisenstat PODC'06**

Population protocols compute precisely predicates definable in Presburger arithmetic, *i.e.* FO($\mathbb{N}$, $+$, $<$)

Verifying correctness

Protocol broken for B = R:

$$B\ R \to b\ b$$
$$B\ r \to B\ b$$
$$R\ b \to R\ r$$

Verifying correctness

Protocol broken for B = R:

$$\boldsymbol{B}\ \boldsymbol{R} \to \boldsymbol{b}\ \boldsymbol{b}$$
$$\boldsymbol{B}\ \boldsymbol{r} \to \boldsymbol{B}\ \boldsymbol{b}$$
$$\boldsymbol{R}\ \boldsymbol{b} \to \boldsymbol{R}\ \boldsymbol{r}$$

B R B R

Verifying correctness

Protocol broken for B = R:

$$B\ R \rightarrow b\ b$$
$$B\ r \rightarrow B\ b$$
$$R\ b \rightarrow R\ r$$

B R B R → B R b b

Verifying correctness

Protocol broken for B = R:

$$\boldsymbol{B}\ \boldsymbol{R} \rightarrow \boldsymbol{b}\ \boldsymbol{b}$$
$$\boldsymbol{B}\ \boldsymbol{r} \rightarrow \boldsymbol{B}\ \boldsymbol{b}$$
$$\boldsymbol{R}\ \boldsymbol{b} \rightarrow \boldsymbol{R}\ \boldsymbol{r}$$

$$\mathbf{B\ R\ B\ R} \rightarrow \mathbf{B\ R\ b\ b} \rightarrow \mathbf{B\ R\ r\ b}$$

Verifying correctness

Protocol broken for B = R:

$$B\ R \rightarrow b\ b$$
$$B\ r \rightarrow B\ b$$
$$R\ b \rightarrow R\ r$$

$$\mathbf{B\ R\ B\ R} \rightarrow \mathbf{B\ R\ b\ b} \rightarrow \mathbf{B\ R\ r\ b} \rightarrow \mathbf{b\ b\ r\ b}$$

Verifying correctness

Protocol correct with tie-breaker:

$$\begin{aligned} B\,R &\rightarrow b\,b \\ B\,r &\rightarrow B\,b \\ R\,b &\rightarrow R\,r \\ b\,r &\rightarrow b\,b \end{aligned}$$

$$\mathbf{B\,R\,B\,R} \rightarrow \mathbf{B\,R\,b\,b} \rightarrow \mathbf{B\,R\,r\,b} \rightarrow \mathbf{b\,b\,r\,b}$$

Verifying correctness

Protocol correct with tie-breaker:

$$\begin{aligned} \boldsymbol{B}\,\boldsymbol{R} &\to \boldsymbol{b}\,\boldsymbol{b} \\ \boldsymbol{B}\,\boldsymbol{r} &\to \boldsymbol{B}\,\boldsymbol{b} \\ \boldsymbol{R}\,\boldsymbol{b} &\to \boldsymbol{R}\,\boldsymbol{r} \\ \boldsymbol{b}\,\boldsymbol{r} &\to \boldsymbol{b}\,\boldsymbol{b} \end{aligned}$$

B R B R $\to$ **B R b b** $\to$ **B R r b** $\to$ **b b r b** $\to$ **b b b b**

Easy fix, but protocols can become complex even for B ≥ R:

Fast and Exact Majority in Population Protocols

Dan Alistarh
Microsoft Research

Rati Gelashvili*
MIT

Milan Vojnović
Microsoft Research

$$weight(x) = \begin{cases} |x| & \text{if } x \in StrongStates \text{ or } x \in WeakStates; \\ 1 & \text{if } x \in IntermediateStates. \end{cases}$$

$$sgn(x) = \begin{cases} 1 & \text{if } x \in \{+0, 1_d, \ldots, 1_1, 3, 5, \ldots, m\}; \\ -1 & \text{otherwise.} \end{cases}$$

$value(x) = sgn(x) \cdot weight(x)$

`/* Functions for rounding state interactions */`

$\phi(x) = -1_1$ if $x = -1$; 1_1 if $x = 1$; x, otherwise

$R_{\downarrow}(k) = \phi(k$ if k odd integer, $k-1$ if k even)

$R_{\uparrow}(k) = \phi(k$ if k odd integer, $k+1$ if k even)

$$Shift\text{-}to\text{-}Zero(x) = \begin{cases} -1_{j+1} & \text{if } x = -1_j \text{ for some index } j < d \\ 1_{j+1} & \text{if } x = 1_j \text{ for some index } j < d \\ x & \text{otherwise.} \end{cases}$$

$$Sign\text{-}to\text{-}Zero(x) = \begin{cases} +0 & \text{if } sgn(x) > 0 \\ -0 & \text{oherwise.} \end{cases}$$

procedure update$\langle x, y\rangle$
  if $(weight(x) > 0$ *and* $weight(y) > 1)$ ***or*** $(weight(y) > 0$ *and* $weight(x) > 1)$ **then**
    $x' \leftarrow R_{\downarrow}\left(\frac{value(x)+value(y)}{2}\right)$ and $y' \leftarrow R_{\uparrow}\left(\frac{value(x)+value(y)}{2}\right)$
  else if $weight(x) \cdot weight(y) = 0$ *and* $value(x) + value(y) > 0$ **then**
    if $weight(x) \neq 0$ **then** $x' \leftarrow Shift\text{-}to\text{-}Zero(x)$ and $y' \leftarrow Sign\text{-}to\text{-}Zero(x)$
    else $y' \leftarrow Shift\text{-}to\text{-}Zero(y)$ and $x' \leftarrow Sign\text{-}to\text{-}Zero(y)$
  else if $(x \in \{-1_d, +1_d\}$ *and* $weight(y) = 1$ *and* $sgn(x) \neq sgn(y))$ ***or***
      $(y \in \{-1_d, +1_d\}$ *and* $weight(x) = 1$ *and* $sgn(y) \neq sgn(x))$ **then**
    $x' \leftarrow -0$ and $y' \leftarrow +0$
  else
    $x' \leftarrow Shift\text{-}to\text{-}Zero(x)$ and $y' \leftarrow Shift\text{-}to\text{-}Zero(y)$

Verifying correctness

Easy fix, but protocols can become complex even for B ≥ R:

Fast and Exact Majority in Population Protocols

Dan Alistarh
Microsoft Research

Rati Gelashvili*
MIT

Milan Vojnović
Microsoft Research

$weight(x) = \begin{cases} |x| & \text{if } x \in StrongStates \text{ or } x \in WeakStates; \\ 1 & \text{if } x \in IntermediateStates. \end{cases}$

$sgn(x) = \begin{cases} 1 & \text{if } x \in \{+0, 1_d, \ldots, 1_1, 3, 5, \ldots, m\}; \\ -1 & \text{otherwise.} \end{cases}$

$value(x) = sgn(x) \cdot weight(x)$

```
/* Functions for rounding state interactions */
```

$\phi(x) = -1_1$ if $x = -1; 1_1$ if $x = 1; x,$ otherwise

$R_\downarrow(k) = \phi(k$ if k odd integer, $k - 1$ if k even$)$

$R_\uparrow(k) = \phi(k$ if k odd integer, $k + 1$ if k even$)$

$Shift\text{-}to\text{-}Zero(x) = \begin{cases} -1_{j+1} & \text{if } x = -1_j \text{ for some index } j < d \\ 1_{j+1} & \text{if } x = 1_j \text{ for some index } j < d \\ x & \text{otherwise.} \end{cases}$

$Sign\text{-}to\text{-}Zero(x) = \begin{cases} +0 & \text{if } sgn(x) > 0 \\ -0 & \text{oherwise.} \end{cases}$

procedure update$\langle x, y \rangle$
  if $(weight(x) > 0$ *and* $weight(y) > 1)$ ***or*** $(weight(y) > 0$ *and* $weight(x) > 1)$ **then**
    $x' \leftarrow R_\downarrow\left(\frac{value(x)+value(y)}{2}\right)$ and $y' \leftarrow R_\uparrow\left(\frac{value(x)+value(y)}{2}\right)$
  else if $weight(x) \cdot weight(y) = 0$ *and* $value(x) + value(y) > 0$ **then**
    if $weight(x) \neq 0$ **then** $x' \leftarrow Shift\text{-}to\text{-}Zero(x)$ and $y' \leftarrow Sign\text{-}to\text{-}Zero(x)$
    else $y' \leftarrow Shift\text{-}to\text{-}Zero(y)$ and $x' \leftarrow Sign\text{-}to\text{-}Zero(y)$
  else if $(x \in \{-1_d, +1_d\}$ *and* $weight(y) = 1$ *and* $sgn(x) \neq sgn(y))$ ***or***
    $(y \in \{-1_d, +1_d\}$ *and* $weight(x) = 1$ *and* $sgn(y) \neq sgn(x))$ **then**
    $x' \leftarrow -0$ and $y' \leftarrow +0$
  else
    $x' \leftarrow Shift\text{-}to\text{-}Zero(x)$ and $y' \leftarrow Shift\text{-}to\text{-}Zero(y)$

How to verify correctness automatically?

Testing whether a protocol computes φ amounts to testing:

$$\begin{aligned}\neg\exists C, D:\quad & C \xrightarrow{*} D\ \wedge \\ & C \text{ is initial} \wedge \\ & D \text{ is in a BSCC} \wedge \\ & \text{opinion}(D) \neq \varphi(C)\end{aligned}$$

Verifying correctness

Testing whether a protocol computes φ amounts to testing:

$$\begin{aligned} \neg\exists C, D:\quad & C \xrightarrow{*} D \,\wedge \\ & C \text{ is initial} \,\wedge \\ & D \text{ is in a BSCC} \,\wedge \\ & \text{opinion}(D) \neq \varphi(C) \end{aligned}$$

Theorem **Esparza *et al.* CONCUR'15**

Verification is decidable

Verification: 1st approach

$$\neg\exists C, D: \quad C \xrightarrow{*} D \wedge$$
$$C \text{ is initial} \wedge$$
$$D \text{ is in a BSCC} \wedge$$
$$\text{opinion}(D) \neq \varphi(C)$$

As difficult as verification

Ackermannan-complete

(Leroux; Czerwinski & Orlikowski FOCS'21, Esparza et al. CONCUR'15)

$$\begin{aligned} \neg\exists C, D:\quad & C \xrightarrow{*} D \wedge \\ & C \text{ is initial } \wedge \\ & D \text{ is in a BSCC } \wedge \\ & \text{opinion}(D) \neq \varphi(C) \end{aligned}$$

Relaxed with Presburger-definable overapproximation!

Verification: 1st approach

$$\neg\exists C, D: \quad C \overset{*}{\dashrightarrow} D \wedge$$

$$C \text{ is initial} \wedge$$

$$D \text{ is in a BSCC} \wedge$$

$$\text{opinion}(D) \neq \varphi(C)$$

Difficult to express

$$\neg\exists C, D: \quad C \overset{*}{\dashrightarrow} D \wedge$$
$$C \text{ is initial} \wedge$$
$$D \text{ is terminal} \wedge$$
$$\text{opinion}(D) \neq \varphi(C)$$

BSCCs are of size 1
for many protocols!

Verification: 1st approach

$$\neg\exists C, D:\quad \begin{aligned} & C \overset{*}{\dashrightarrow} D \,\wedge \\ & C \text{ is initial} \wedge \\ & D \text{ is terminal } \wedge \\ & \text{opinion}(D) \neq \varphi(C) \end{aligned}$$

Testable with an SMT solver

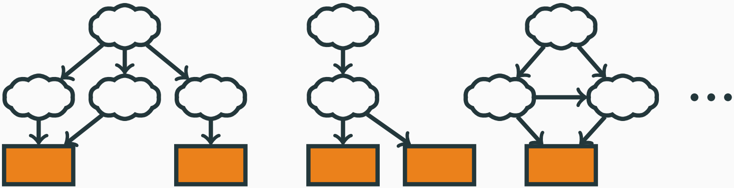
Verification: 1st approach

$$\begin{aligned} \neg\exists C, D:\ & C \overset{*}{\dashrightarrow} D \wedge \\ & C \text{ is initial} \wedge \\ & D \text{ is terminal} \wedge \\ & \text{opinion}(D) \neq \varphi(C) \end{aligned}$$

But how to know whether all BSCCs are of size 1?

Silent protocols

A protocol is *silent* if fair executions reach terminal configurations

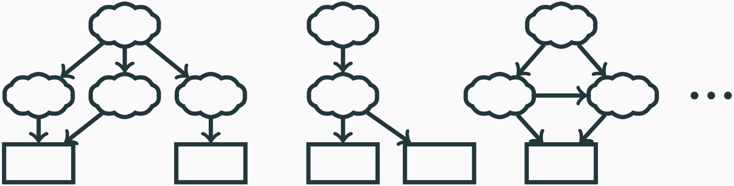


BSCCs of size 1

Silent protocols

A protocol is *silent* if fair executions reach terminal configurations

- Testing silentness is as hard as verification of correctness
- But many protocols satisfy a common design

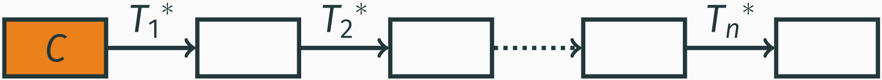


BSCCs of size 1

Silent protocols: layered termination

Partition $T = T_1 \cup T_2 \cup \cdots \cup T_n$ s.t. for every i

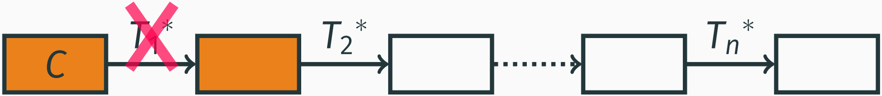
- all executions restricted to T_i terminate
- if $T_1 \cup \cdots \cup T_{i-1}$ disabled in C and $C \xrightarrow{T_i{}^*} D$, then $T_1 \cup \cdots \cup T_{i-1}$ also disabled in D



Silent protocols: layered termination

Partition $T = T_1 \cup T_2 \cup \cdots \cup T_n$ s.t. for every i

- all executions restricted to T_i terminate
- if $T_1 \cup \cdots \cup T_{i-1}$ disabled in C and $C \xrightarrow{T_i^*} D$, then $T_1 \cup \cdots \cup T_{i-1}$ also disabled in D



Silent protocols: layered termination

Partition $T = T_1 \cup T_2 \cup \cdots \cup T_n$ s.t. for every i

- all executions restricted to T_i terminate
- if $T_1 \cup \cdots \cup T_{i-1}$ disabled in C and $C \xrightarrow{T_i^*} D$, then $T_1 \cup \cdots \cup T_{i-1}$ also disabled in D

Silent protocols: layered termination

Partition $T = T_1 \cup T_2 \cup \cdots \cup T_n$ s.t. for every i

- all executions restricted to T_i terminate
- if $T_1 \cup \cdots \cup T_{i-1}$ disabled in C and $C \xrightarrow{T_i^*} D$, then $T_1 \cup \cdots \cup T_{i-1}$ also disabled in D

Silent protocols: layered termination

Partition $T = T_1 \cup T_2 \cup \cdots \cup T_n$ s.t. for every i

- all executions restricted to T_i terminate
- if $T_1 \cup \cdots \cup T_{i-1}$ disabled in C and $C \xrightarrow{T_i{}^*} D$, then $T_1 \cup \cdots \cup T_{i-1}$ also disabled in D

Silent protocols: layered termination

T_1

$$\begin{aligned} B\ R &\to b\ b \\ B\ r &\to B\ b \\ R\ b &\to R\ r \\ b\ r &\to b\ b \end{aligned}$$

Silent protocols: layered termination

T_1

$$\begin{aligned} \boldsymbol{B}\ \boldsymbol{R} &\rightarrow \boldsymbol{b}\ \boldsymbol{b} \\ \boldsymbol{B}\ \boldsymbol{r} &\rightarrow \boldsymbol{B}\ \boldsymbol{b} \\ \boldsymbol{R}\ \boldsymbol{b} &\rightarrow \boldsymbol{R}\ \boldsymbol{r} \\ \boldsymbol{b}\ \boldsymbol{r} &\rightarrow \boldsymbol{b}\ \boldsymbol{b} \end{aligned}$$

Bad partition: not all executions over T_1 terminate

Silent protocols: layered termination

T_1

$$\begin{aligned} B\ R &\to b\ b \\ \boxed{B\ r} &\to \boxed{B\ b} \\ \boxed{R\ b} &\to \boxed{R\ r} \\ b\ r &\to b\ b \end{aligned}$$

Bad partition: not all executions over T_1 terminate

$$\{B, B, R, R\} \to \{B, b, b, R\} \to \{B, b, r, R\} \to \{B, b, b, R\} \to \{B, b, r, R\} \to \cdots$$

Silent protocols: layered termination

T_1

$$\boldsymbol{B}\ \boldsymbol{R} \rightarrow \boldsymbol{b}\ \boldsymbol{b}$$

T_2

$$\boldsymbol{R}\ \boldsymbol{b} \rightarrow \boldsymbol{R}\ \boldsymbol{r}$$

T_3

$$\boldsymbol{B}\ \boldsymbol{r} \rightarrow \boldsymbol{B}\ \boldsymbol{b}$$

$$\boldsymbol{b}\ \boldsymbol{r} \rightarrow \boldsymbol{b}\ \boldsymbol{b}$$

Silent protocols: layered termination

T_1

$$\boldsymbol{B}\ \boldsymbol{R} \rightarrow \boldsymbol{b}\ \boldsymbol{b}$$

T_2

$$\boldsymbol{R}\ \boldsymbol{b} \rightarrow \boldsymbol{R}\ \boldsymbol{r}$$

T_3

$$\boldsymbol{B}\ \boldsymbol{r} \rightarrow \boldsymbol{B}\ \boldsymbol{b}$$

$$\boldsymbol{b}\ \boldsymbol{r} \rightarrow \boldsymbol{b}\ \boldsymbol{b}$$

$\#\boldsymbol{B} \geq \#\boldsymbol{R}$:

$$\{\boldsymbol{B}^*,\ \boldsymbol{R}^*\}$$

Silent protocols: layered termination

T_1 ✗

$$B\ R \to b\ b$$

T_2

$$R\ b \to R\ r$$

T_3

$$B\ r \to B\ b$$

$$b\ r \to b\ b$$

$\#B \geq \#R$:

$$\{B^*,\ R^*\} \xrightarrow{*} \{B^*,\ b^*,\ r^*\}$$

Silent protocols: layered termination

T_1 ✗

$$B\ R \rightarrow b\ b$$

T_2 ✗

$$R\ b \rightarrow R\ r$$

T_3

$$B\ r \rightarrow B\ b$$

$$b\ r \rightarrow b\ b$$

$\#B \geq \#R$:

$$\{B^*,\ R^*\} \xrightarrow{*} \{B^*,\ b^*,\ r^*\}$$

Silent protocols: layered termination

T_1 ✗

$$B\ R \rightarrow b\ b$$

T_2 ✗

$$R\ b \rightarrow R\ r$$

T_3 ✗

$$B\ r \rightarrow B\ b$$
$$b\ r \rightarrow b\ b$$

$\#B \geq \#R$:

$$\{B^*,\ R^*\} \xrightarrow{*} \{B^*,\ b^*,\ r^*\} \xrightarrow{*} \{B^*,\ b^*\}$$

Silent protocols: layered termination

T_1

$$B\ R \to b\ b$$

T_2

$$R\ b \to R\ r$$

T_3

$$B\ r \to B\ b$$
$$b\ r \to b\ b$$

$\#B \geq \#R$:

$$\{B^*,\ R^*\} \xrightarrow{*} \{B^*,\ b^*,\ r^*\} \xrightarrow{*} \{B^*,\ b^*\}$$

$\#R > \#B$:

$$\{R^+,\ B^*\}$$

Silent protocols: layered termination

T_1 ✗

$$\boldsymbol{B}\ \boldsymbol{R} \to \boldsymbol{b}\ \boldsymbol{b}$$

T_2

$$\boldsymbol{R}\ \boldsymbol{b} \to \boldsymbol{R}\ \boldsymbol{r}$$

T_3

$$\boldsymbol{B}\ \boldsymbol{r} \to \boldsymbol{B}\ \boldsymbol{b}$$

$$\boldsymbol{b}\ \boldsymbol{r} \to \boldsymbol{b}\ \boldsymbol{b}$$

$\#\boldsymbol{B} \geq \#\boldsymbol{R}$:

$$\{\boldsymbol{B}^*,\ \boldsymbol{R}^*\} \xrightarrow{*} \{\boldsymbol{B}^*,\ \boldsymbol{b}^*,\ \boldsymbol{r}^*\} \xrightarrow{*} \{\boldsymbol{B}^*,\ \boldsymbol{b}^*\}$$

$\#\boldsymbol{R} > \#\boldsymbol{B}$:

$$\{\boldsymbol{R}^+,\ \boldsymbol{B}^*\} \xrightarrow{*} \{\boldsymbol{R}^+,\ \boldsymbol{b}^*, \boldsymbol{r}^*\}$$

Silent protocols: layered termination

T_1 ✗

$$B\ R \to b\ b$$

T_2 ✗

$$R\ b \to R\ r$$

T_3

$$B\ r \to B\ b$$
$$b\ r \to b\ b$$

$\#B \geq \#R$:

$$\{B^*,\ R^*\} \xrightarrow{*} \{B^*,\ b^*,\ r^*\} \xrightarrow{*} \{B^*,\ b^*\}$$

$\#R > \#B$:

$$\{R^+,\ B^*\} \xrightarrow{*} \{R^+,\ b^*, r^*\} \xrightarrow{*} \{R^+,\ r^*\}$$

Silent protocols: layered termination

T_1 ✗

$B\ R \to b\ b$

T_2 ✗

$R\ b \to R\ r$

T_3 ✗

$B\ r \to B\ b$

$b\ r \to b\ b$

$\#B \geq \#R$:

$$\{B^*,\ R^*\} \xrightarrow{*} \{B^*,\ b^*,\ r^*\} \xrightarrow{*} \{B^*,\ b^*\}$$

$\#R > \#B$:

$$\{R^+,\ B^*\} \xrightarrow{*} \{R^+,\ b^*, r^*\} \xrightarrow{*} \{R^+,\ r^*\}$$

Silent protocols: layered termination

T_1

$$\mathbf{B}\,\mathbf{R} \to \mathbf{b}\,\mathbf{b}$$

T_2

$$\mathbf{R}\,\mathbf{b} \to \mathbf{R}\,\mathbf{r}$$

T_3

$$\mathbf{B}\,\mathbf{r} \to \mathbf{B}\,\mathbf{b}$$

$$\mathbf{b}\,\mathbf{r} \to \mathbf{b}\,\mathbf{b}$$

Theorem

Deciding whether a protocol is strongly silent $\in$ NP

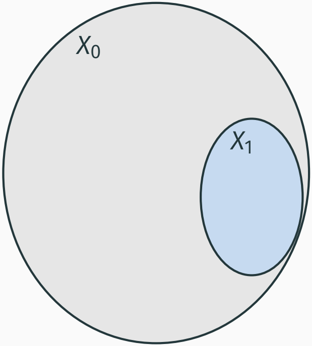
Recent efficient protocols are not silent!

Recent efficient protocols are not silent!

More powerful approach:
using “correctness certificates”

Correctness certificates

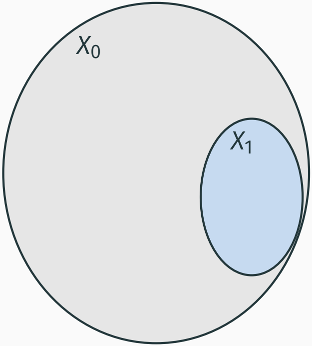
Approach: certify that φ is computed correctly for $b \in \{0, 1\}$



$r_0 \colon \text{Configs} \to \mathbb{N}$

Correctness certificates

Approach: certify that φ is computed correctly for $b \in \{0, 1\}$



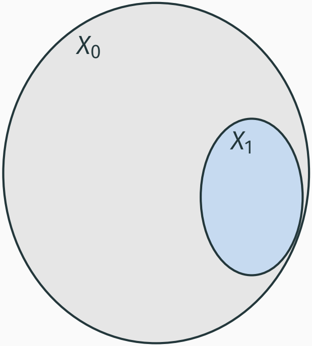
r_0 : Configs $\to \mathbb{N}$

- $C \in X_i \land C \xrightarrow{*} C' \implies C' \in X_i$

Closed under reachability

Correctness certificates

Approach: certify that φ is computed correctly for $b \in \{0, 1\}$



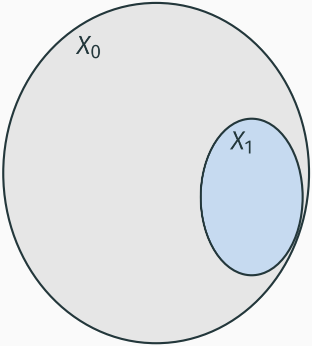
$r_0 :$ Configs $\to \mathbb{N}$

- $C \in X_i \wedge C \xrightarrow{*} C' \implies C' \in X_i$
- $X_0 \supseteq \{C : C \text{ is initial and } \varphi(C) = b\}$

X0 contains all initial configs

Correctness certificates

Approach: certify that φ is computed correctly for $b \in \{0, 1\}$



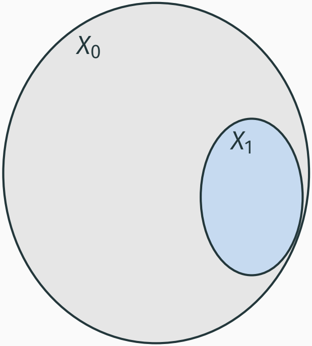
r_0 : Configs $\to \mathbb{N}$

- $C \in X_i \wedge C \xrightarrow{*} C' \implies C' \in X_i$
- $X_0 \supseteq \{C : C \text{ is initial and } \varphi(C) = b\}$
- $X_1 \subseteq \{C : \text{opinion}(C) = b\}$

X_1 only contains configs with b-consensus

Correctness certificates

Approach: certify that φ is computed correctly for $b \in \{0, 1\}$



$r_0 : \text{Configs} \to \mathbb{N}$

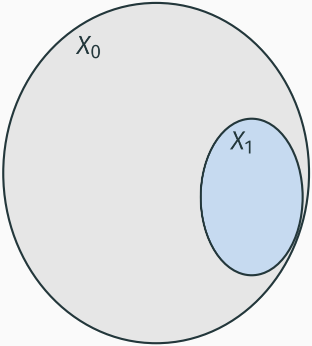
- $C \in X_i \wedge C \xrightarrow{*} C' \implies C' \in X_i$
- $X_0 \supseteq \{C : C \text{ is initial and } \varphi(C) = b\}$
- $X_1 \subseteq \{C : \text{opinion}(C) = b\}$

- $C \xrightarrow{*} C' \implies r_0(C) \geq r_0(C')$

r_0 is nondecreasing

Correctness certificates

Approach: certify that φ is computed correctly for $b \in \{0, 1\}$



$r_0 : \text{Configs} \to \mathbb{N}$

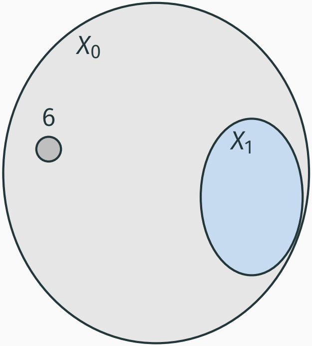
- $C \in X_i \wedge C \xrightarrow{*} C' \implies C' \in X_i$
- $X_0 \supseteq \{C : C \text{ is initial and } \varphi(C) = b\}$
- $X_1 \subseteq \{C : \text{opinion}(C) = b\}$

- $C \xrightarrow{*} C' \implies r_0(C) \geq r_0(C')$
- $\forall C \in X_0 \setminus X_1 \, \exists C' \in X_0 : C \xrightarrow{*} C' \wedge r_0(C) > r_0(C')$

r0 is weakly decreasing

Correctness certificates

Approach: certify that φ is computed correctly for $b \in \{0, 1\}$



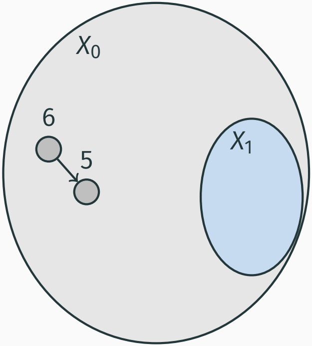
r_0 : Configs $\to \mathbb{N}$

- $C \in X_i \wedge C \xrightarrow{*} C' \implies C' \in X_i$
- $X_0 \supseteq \{C : C \text{ is initial and } \varphi(C) = b\}$
- $X_1 \subseteq \{C : \text{opinion}(C) = b\}$

- $C \xrightarrow{*} C' \implies r_0(C) \geq r_0(C')$
- $\forall C \in X_0 \setminus X_1 \, \exists C' \in X_0 : C \xrightarrow{*} C' \wedge r_0(C) > r_0(C')$

Correctness certificates

Approach: certify that φ is computed correctly for $b \in \{0, 1\}$



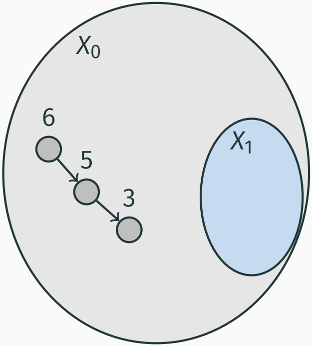
r_0 : Configs $\to \mathbb{N}$

- $C \in X_i \wedge C \xrightarrow{*} C' \implies C' \in X_i$
- $X_0 \supseteq \{C : C \text{ is initial and } \varphi(C) = b\}$
- $X_1 \subseteq \{C : \text{opinion}(C) = b\}$

- $C \xrightarrow{*} C' \implies r_0(C) \geq r_0(C')$
- $\forall C \in X_0 \setminus X_1 \, \exists C' \in X_0 : C \xrightarrow{*} C' \wedge r_0(C) > r_0(C')$

Correctness certificates

Approach: certify that φ is computed correctly for $b \in \{0,1\}$



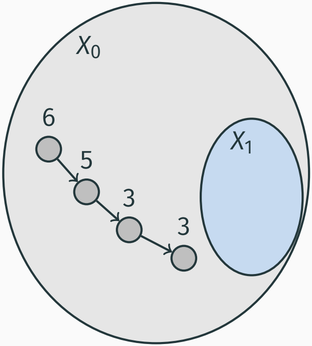
r_0 : Configs $\to \mathbb{N}$

- $C \in X_i \wedge C \xrightarrow{*} C' \implies C' \in X_i$
- $X_0 \supseteq \{C : C \text{ is initial and } \varphi(C) = b\}$
- $X_1 \subseteq \{C : \text{opinion}(C) = b\}$

- $C \xrightarrow{*} C' \implies r_0(C) \geq r_0(C')$
- $\forall C \in X_0 \setminus X_1 \, \exists C' \in X_0 : C \xrightarrow{*} C' \wedge r_0(C) > r_0(C')$

Correctness certificates

Approach: certify that φ is computed correctly for $b \in \{0, 1\}$



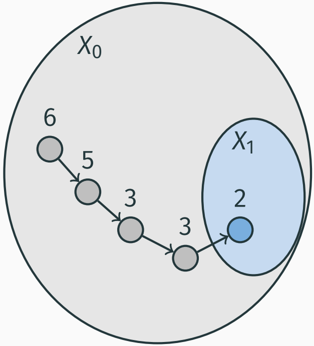
r_0 : Configs $\to \mathbb{N}$

- $C \in X_i \wedge C \xrightarrow{*} C' \implies C' \in X_i$
- $X_0 \supseteq \{C : C \text{ is initial and } \varphi(C) = b\}$
- $X_1 \subseteq \{C : \text{opinion}(C) = b\}$

- $C \xrightarrow{*} C' \implies r_0(C) \geq r_0(C')$
- $\forall C \in X_0 \setminus X_1 \, \exists C' \in X_0 : C \xrightarrow{*} C' \wedge r_0(C) > r_0(C')$

Correctness certificates

Approach: certify that φ is computed correctly for $b \in \{0, 1\}$



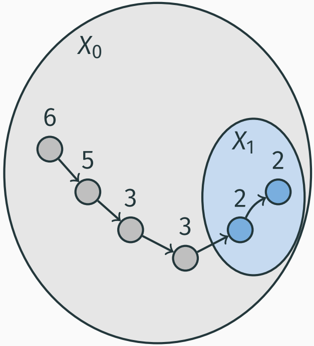
r_0 : Configs $\to \mathbb{N}$

- $C \in X_i \wedge C \xrightarrow{*} C' \implies C' \in X_i$
- $X_0 \supseteq \{C : C \text{ is initial and } \varphi(C) = b\}$
- $X_1 \subseteq \{C : \text{opinion}(C) = b\}$

- $C \xrightarrow{*} C' \implies r_0(C) \geq r_0(C')$
- $\forall C \in X_0 \setminus X_1 \, \exists C' \in X_0 : C \xrightarrow{*} C' \wedge r_0(C) > r_0(C')$

Correctness certificates

Approach: certify that φ is computed correctly for $b \in \{0,1\}$



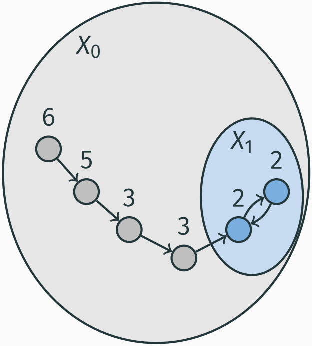
r_0 : Configs $\to \mathbb{N}$

- $C \in X_i \wedge C \xrightarrow{*} C' \implies C' \in X_i$
- $X_0 \supseteq \{C : C \text{ is initial and } \varphi(C) = b\}$
- $X_1 \subseteq \{C : \text{opinion}(C) = b\}$

- $C \xrightarrow{*} C' \implies r_0(C) \geq r_0(C')$
- $\forall C \in X_0 \setminus X_1\, \exists C' \in X_0 : C \xrightarrow{*} C' \wedge r_0(C) > r_0(C')$

Correctness certificates

Approach: certify that φ is computed correctly for $b \in \{0,1\}$



r_0: Configs $\to \mathbb{N}$

- $C \in X_i \wedge C \xrightarrow{*} C' \implies C' \in X_i$
- $X_0 \supseteq \{C : C \text{ is initial and } \varphi(C) = b\}$
- $X_1 \subseteq \{C : \text{opinion}(C) = b\}$

- $C \xrightarrow{*} C' \implies r_0(C) \geq r_0(C')$
- $\forall C \in X_0 \setminus X_1 \, \exists C' \in X_0 : C \xrightarrow{*} C' \wedge r_0(C) > r_0(C')$

Stage graphs

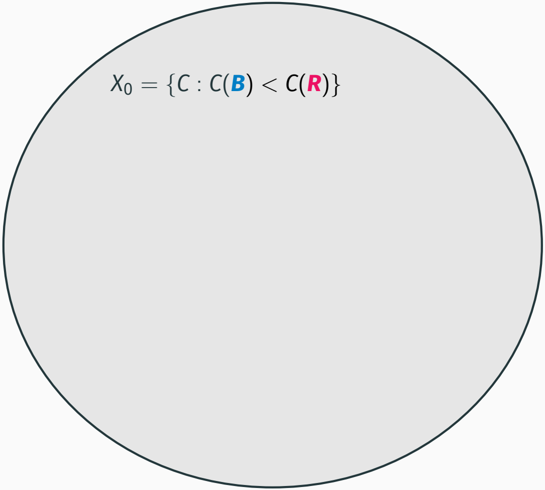
Stage graph: same idea with $X_0, X_1, \ldots, X_k$ organized in a DAG

$$
\begin{aligned}
\mathbf{B}\,\mathbf{R} &\to \mathbf{b}\,\mathbf{b}\\
\mathbf{B}\,\mathbf{r} &\to \mathbf{B}\,\mathbf{b}\\
\mathbf{R}\,\mathbf{b} &\to \mathbf{R}\,\mathbf{r}\\
\mathbf{b}\,\mathbf{r} &\to \mathbf{b}\,\mathbf{b}
\end{aligned}
$$

Stage graphs

Stage graph: same idea with $X_0, X_1, \ldots, X_k$ organized in a DAG

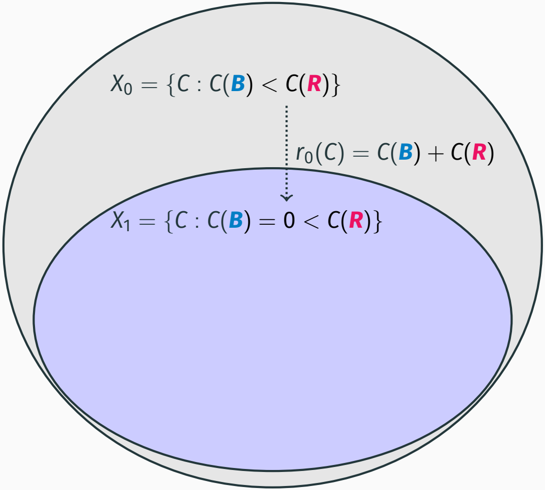
$$\mathbf{B}\ \mathbf{R} \rightarrow \mathbf{b}\ \mathbf{b}$$
$$\mathbf{B}\ \mathbf{r} \rightarrow \mathbf{B}\ \mathbf{b}$$
$$\mathbf{R}\ \mathbf{b} \rightarrow \mathbf{R}\ \mathbf{r}$$
$$\mathbf{b}\ \mathbf{r} \rightarrow \mathbf{b}\ \mathbf{b}$$



Stage graphs

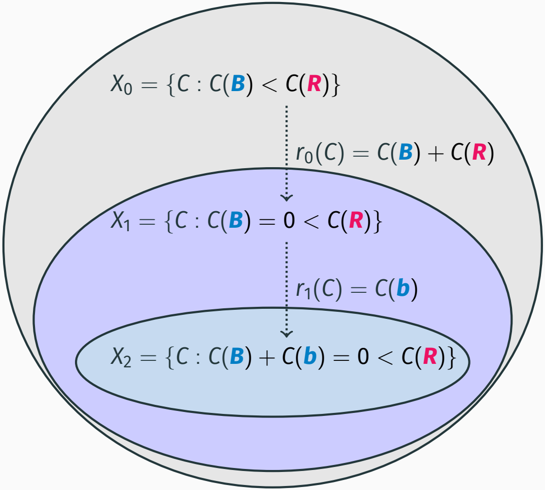
Stage graph: same idea with $X_0, X_1, \ldots, X_k$ organized in a DAG

$$\begin{aligned} B\,R &\to b\,b \\ B\,r &\to B\,b \\ R\,b &\to R\,r \\ b\,r &\to b\,b \end{aligned}$$



Stage graph: same idea with $X_0, X_1, \ldots, X_k$ organized in a DAG

$$B\;R \to b\;b$$
$$B\;r \to B\;b$$
$$R\;b \to R\;r$$
$$b\;r \to b\;b$$



Stage graphs

A stage graph is *Presburger* if

- Each set X_i is Presburger-definable
- Each ranking function r_i is Presburger-definable
- Each r_i can be decreased in at most B_i steps

Stage graphs

A stage graph is *Presburger* if

- Each set X_i is Presburger-definable
- Each ranking function r_i is Presburger-definable
- Each r_i can be decreased in at most B_i steps

Stage graphs

A stage graph is *Presburger* if

- Each set X_i is Presburger-definable
- Each ranking function r_i is Presburger-definable
- Each r_i can be decreased in at most B_i steps

Stage graphs

A stage graph is *Presburger* if

- Each set X_i is Presburger-definable
- Each ranking function r_i is Presburger-definable
- Each r_i can be decreased in at most B_i steps

Stage graphs

A stage graph is *Presburger* if

- Each set X_i is Presburger-definable
- Each ranking function r_i is Presburger-definable
- Each r_i can be decreased in at most B_i steps

Theorem

Every correct protocol has Presburger stage graphs

Stage graphs

A stage graph is *Presburger* if

- Each set X_i is Presburger-definable
- Each ranking function r_i is Presburger-definable
- Each r_i can be decreased in at most B_i steps

Theorem

Every correct protocol has Presburger stage graphs

Computable and checkable in practice with SMT solving!

Demonstration

Expected termination time

$\mathbf{B}, \mathbf{R} \mapsto \mathbf{b}, \mathbf{b}$

$\mathbf{B}, \mathbf{r} \mapsto \mathbf{B}, \mathbf{b}$

$\mathbf{R}, \mathbf{b} \mapsto \mathbf{R}, \mathbf{r}$

$\mathbf{b}, \mathbf{r} \mapsto \mathbf{b}, \mathbf{b}$

Correctly computes predicate $\#\mathbf{B} \geq \#\mathbf{R}$

...but how fast?

Expected termination time

$\mathbf{B}, \mathbf{R} \mapsto \mathbf{b}, \mathbf{b}$

$\mathbf{B}, \mathbf{r} \mapsto \mathbf{B}, \mathbf{b}$

$\mathbf{R}, \mathbf{b} \mapsto \mathbf{R}, \mathbf{r}$

$\mathbf{b}, \mathbf{r} \mapsto \mathbf{b}, \mathbf{b}$

Correctly computes predicate $\#\mathbf{B} \geq \#\mathbf{R}$

...but how fast?

- **Natural to look for fast protocols**
- **Bounds on expected termination time useful since generally not possible to know whether a protocol has stabilized**

Expected termination time

$\mathbf{B}, \mathbf{R} \mapsto \mathbf{b}, \mathbf{b}$

$\mathbf{B}, \mathbf{r} \mapsto \mathbf{B}, \mathbf{b}$

$\mathbf{R}, \mathbf{b} \mapsto \mathbf{R}, \mathbf{r}$

$\mathbf{b}, \mathbf{r} \mapsto \mathbf{b}, \mathbf{b}$

Correctly computes predicate $\#\mathbf{B} \geq \#\mathbf{R}$

...but how fast?

Theorem **Angluin *et al.* PODC'04**

Every Presburger-definable predicate is computable by a protocol with expected termination time $\in \mathcal{O}(n^2 \log n)$

Expected termination time

$$\mathbf{B}, \mathbf{R} \mapsto \mathbf{b}, \mathbf{b}$$
$$\mathbf{B}, \mathbf{r} \mapsto \mathbf{B}, \mathbf{b}$$
$$\mathbf{R}, \mathbf{b} \mapsto \mathbf{R}, \mathbf{r}$$
$$\mathbf{b}, \mathbf{r} \mapsto \mathbf{b}, \mathbf{b}$$

Simulations show that it is slow when **R** has slight majority:

Steps	Initial configuration
100000	{B: 7, R: 8}
7	{B: 3, R: 12}
27	{B: 4, R: 11}
100000	{B: 7, R: 8}
3	{B: 13, R: 2}

Expected termination time

$$\begin{aligned}
\mathbf{B}, \mathbf{R} &\mapsto \mathbf{T}, \mathbf{t} & X, y &\mapsto X, x \quad \text{for } x, y \in \{\mathbf{b}, \mathbf{r}, \mathbf{t}\} \\
\mathbf{B}, \mathbf{T} &\mapsto \mathbf{B}, \mathbf{b} \\
\mathbf{R}, \mathbf{T} &\mapsto \mathbf{R}, \mathbf{r} \\
\mathbf{T}, \mathbf{T} &\mapsto \mathbf{T}, \mathbf{t}
\end{aligned}$$

$$O(\mathbf{B}) = O(\mathbf{b}) = O(\mathbf{T}) = O(\mathbf{t}) = 1$$
$$O(\mathbf{R}) = O(\mathbf{r}) = 0$$

Alternative protocol with explicit ties

Expected termination time

$$\mathbf{B}, \mathbf{R} \mapsto \mathbf{T}, \mathbf{t} \qquad X, y \mapsto X, x \quad \text{for } x, y \in \{\mathbf{b}, \mathbf{r}, \mathbf{t}\}$$

$$\mathbf{B}, \mathbf{T} \mapsto \mathbf{B}, \mathbf{b}$$

$$\mathbf{R}, \mathbf{T} \mapsto \mathbf{R}, \mathbf{r}$$

$$\mathbf{T}, \mathbf{T} \mapsto \mathbf{T}, \mathbf{t}$$

Is it faster?

$$O(\mathbf{B}) = O(\mathbf{b}) = O(\mathbf{T}) = O(\mathbf{t}) = 1$$

$$O(\mathbf{R}) = O(\mathbf{r}) = 0$$

Alternative protocol with explicit ties

Expected termination time

$\mathbf{B}, \mathbf{R} \mapsto \mathbf{T}, \mathbf{t}$

$\mathbf{B}, \mathbf{T} \mapsto \mathbf{B}, \mathbf{b}$

$\mathbf{R}, \mathbf{T} \mapsto \mathbf{R}, \mathbf{r}$

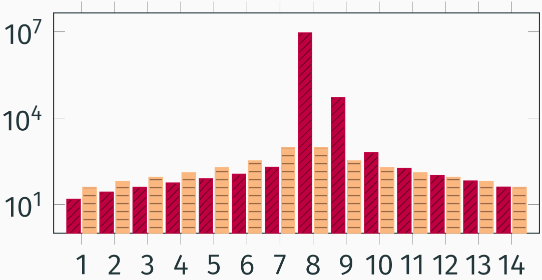
$\mathbf{T}, \mathbf{T} \mapsto \mathbf{T}, \mathbf{t}$

$X, y \mapsto X, x$ for $x, y \in \{\mathbf{b}, \mathbf{r}, \mathbf{t}\}$

Is it faster?

Yes, for size 15...

expected number of steps to stable consensus



number of agents initially in state **R**

Expected termination time

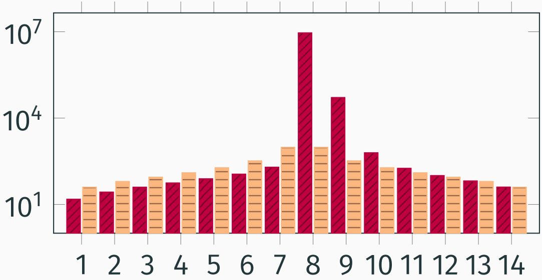
$$\begin{aligned} \mathbf{B}, \mathbf{R} &\mapsto \mathbf{T}, \mathbf{t} \\ \mathbf{B}, \mathbf{T} &\mapsto \mathbf{B}, \mathbf{b} \\ \mathbf{R}, \mathbf{T} &\mapsto \mathbf{R}, \mathbf{r} \\ \mathbf{T}, \mathbf{T} &\mapsto \mathbf{T}, \mathbf{t} \end{aligned}$$

$$X, y \mapsto X, x \quad \text{for } x, y \in \{\mathbf{b}, \mathbf{r}, \mathbf{t}\}$$

Obtained using PRISM

Clément et al. ICDCS'11, Offtermatt '17

expected number of steps to stable consensus



number of agents initially in state **R**

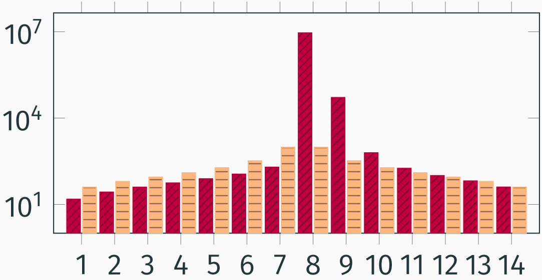
Expected termination time

$$\mathbf{B}, \mathbf{R} \mapsto \mathbf{T}, \mathbf{t}$$
$$\mathbf{B}, \mathbf{T} \mapsto \mathbf{B}, \mathbf{b}$$
$$\mathbf{R}, \mathbf{T} \mapsto \mathbf{R}, \mathbf{r}$$
$$\mathbf{T}, \mathbf{T} \mapsto \mathbf{T}, \mathbf{t}$$

$$X, y \mapsto X, x \quad \text{for } x, y \in \{\mathbf{b}, \mathbf{r}, \mathbf{t}\}$$

Goal: analyze time for all sizes

expected number of steps to stable consensus



number of agents initially in state **R**

Expected termination time: formal definition

Random variable $Steps_X$:

assigns to each run σ the smallest k s.t. $\sigma_k \in X$, otherwise ∞

Expected termination time: formal definition

Random variable Steps_X:

assigns to each run σ the smallest k s.t. $\sigma_k \in X$, otherwise ∞

Maximal expected termination time

We are interested in $\mathit{time}\colon \mathbb{N} \to \mathbb{N}$ where

$$\mathit{time}(n) = \max\{\mathbb{E}_C[\mathit{Steps}_{\text{Stable}}] : C \text{ is initial and } |C| = n\}$$

Expected termination time: formal definition

Random variable $Steps_X$:

assigns to each run σ the smallest k s.t. $\sigma_k \in X$, otherwise ∞

Maximal expected termination time

We are interested in $time \colon \mathbb{N} \to \mathbb{N}$ where

$$time(n) = \max\{\mathbb{E}_C[Steps_{\text{Stable}}] : C \text{ is initial and } |C| = n\}$$

Expected termination time: formal definition

Random variable Steps_X:

assigns to each run σ the smallest k s.t. $\sigma_k \in X$, otherwise ∞

Maximal expected termination time

We are interested in $\mathit{time} \colon \mathbb{N} \to \mathbb{N}$ where

$$\mathit{time}(n) = \max\{\mathbb{E}_C[\mathit{Steps}_{\mathrm{Stable}}] : C \text{ is initial and } |C| = n\}$$

Expected termination time: formal definition

Random variable $Steps_X$:

assigns to each run σ the smallest k s.t. $\sigma_k \in X$, otherwise ∞

Maximal expected termination time

We are interested in $time : \mathbb{N} \to \mathbb{N}$ where

$$time(n) = \max\{\mathbb{E}_C[Steps_{\text{Stable}}] : C \text{ is initial and } |C| = n\}$$

Expected termination time: stage graphs

$$\begin{aligned}
\mathbf{B}, \mathbf{R} &\mapsto \mathbf{T}, \mathbf{t} \\
\mathbf{B}, \mathbf{T} &\mapsto \mathbf{B}, \mathbf{b} \\
\mathbf{R}, \mathbf{T} &\mapsto \mathbf{R}, \mathbf{r} \\
\mathbf{T}, \mathbf{T} &\mapsto \mathbf{T}, \mathbf{t} \\
X, y &\mapsto X, x
\end{aligned}$$

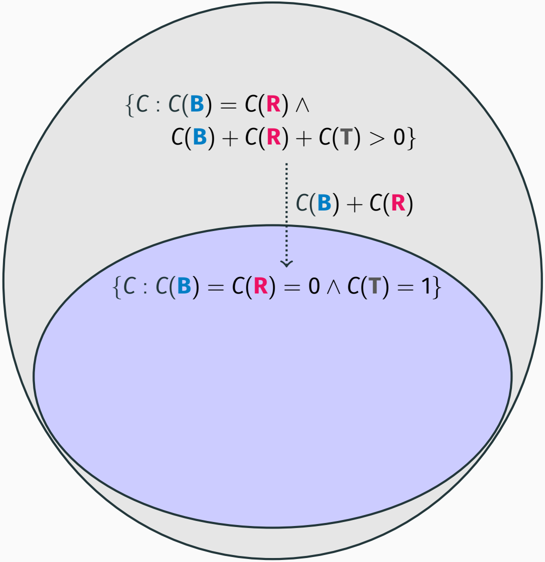
Expected termination time: stage graphs

$$\mathbf{B}, \mathbf{R} \mapsto \mathbf{T}, \mathbf{t}$$
$$\mathbf{B}, \mathbf{T} \mapsto \mathbf{B}, \mathbf{b}$$
$$\mathbf{R}, \mathbf{T} \mapsto \mathbf{R}, \mathbf{r}$$
$$\mathbf{T}, \mathbf{T} \mapsto \mathbf{T}, \mathbf{t}$$
$$X, y \mapsto X, x$$

$$\{C : C(\mathbf{B}) = C(\mathbf{R}) \wedge C(\mathbf{B}) + C(\mathbf{R}) + C(\mathbf{T}) > 0\}$$

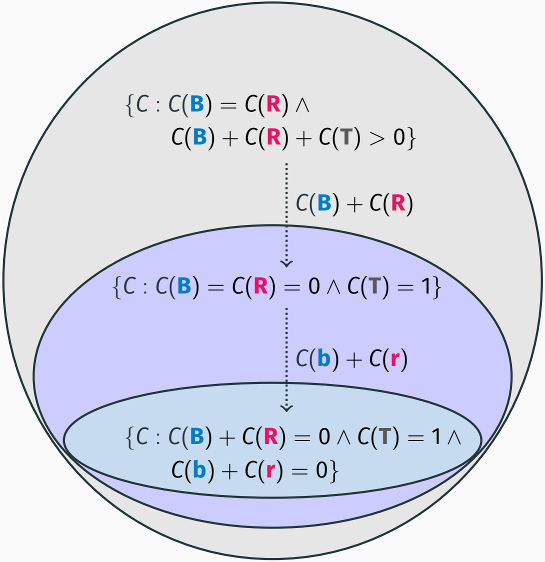
Expected termination time: stage graphs

$$
\begin{aligned}
\mathbf{B}, \mathbf{R} &\mapsto \mathbf{T}, \mathbf{t} \\
\mathbf{B}, \mathbf{T} &\mapsto \mathbf{B}, \mathbf{b} \\
\mathbf{R}, \mathbf{T} &\mapsto \mathbf{R}, \mathbf{r} \\
\mathbf{T}, \mathbf{T} &\mapsto \mathbf{T}, \mathbf{t} \\
X, y &\mapsto X, x
\end{aligned}
$$



Expected termination time: stage graphs

$$
\begin{array}{rcl}
\mathbf{B}, \mathbf{R} & \mapsto & \mathbf{T}, \mathbf{t} \\
\mathbf{B}, \mathbf{T} & \mapsto & \mathbf{B}, \mathbf{b} \\
\mathbf{R}, \mathbf{T} & \mapsto & \mathbf{R}, \mathbf{r} \\
\mathbf{T}, \mathbf{T} & \mapsto & \mathbf{T}, \mathbf{t} \\
X, y & \mapsto & X, x
\end{array}
$$



Expected termination time: stage graphs

$$\mathbf{B}, \mathbf{R} \mapsto \mathbf{T}, \mathbf{t}$$

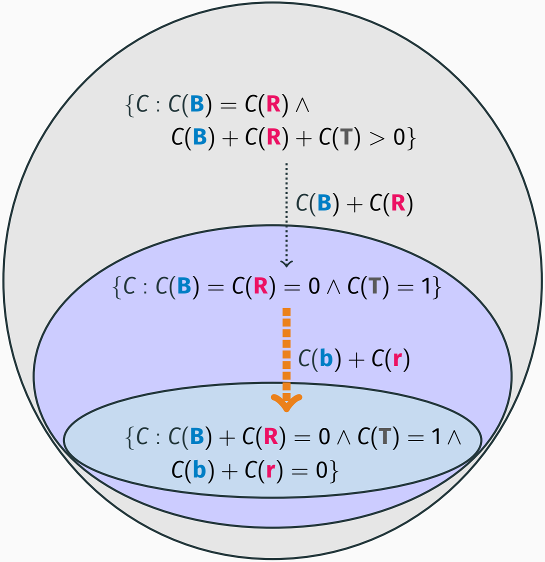
$$\mathbf{B}, \mathbf{T} \mapsto \mathbf{B}, \mathbf{b}$$

$$\mathbf{R}, \mathbf{T} \mapsto \mathbf{R}, \mathbf{r}$$

$$\mathbf{T}, \mathbf{T} \mapsto \mathbf{T}, \mathbf{t}$$

$$X, y \mapsto X, x$$

$$\begin{aligned}\mathbb{E}_C[Steps_{C(\mathbf{b})+C(\mathbf{r})=0}] &\leq \sum_{i=1}^{C(\mathbf{b})+C(\mathbf{r})} \frac{n^2}{2 \cdot C(\mathbf{T}) \cdot i} \\ &\leq \sum_{i=1}^{n} \frac{n^2}{i} \\ &\leq \alpha \cdot n^2 \cdot \log n\end{aligned}$$



In practice, able to report:

$$\mathcal{O}(n^2),\ \mathcal{O}(n^2 \log n),\ \mathcal{O}(n^3),\ \mathcal{O}(n^c),\ \mathcal{O}(2^n)$$

Demonstration

Population protocols analyzable automatically:

- Verification + explanation of correctness
- Bounds on expected termination time
- Tool support

Conclusion: future work

- Asymptotic *lower* bounds on expected termination time?
- Verification of extensions of the model?
- Quantitative model checking?

Thank you!