Constant-space population protocols for uniform bipartition

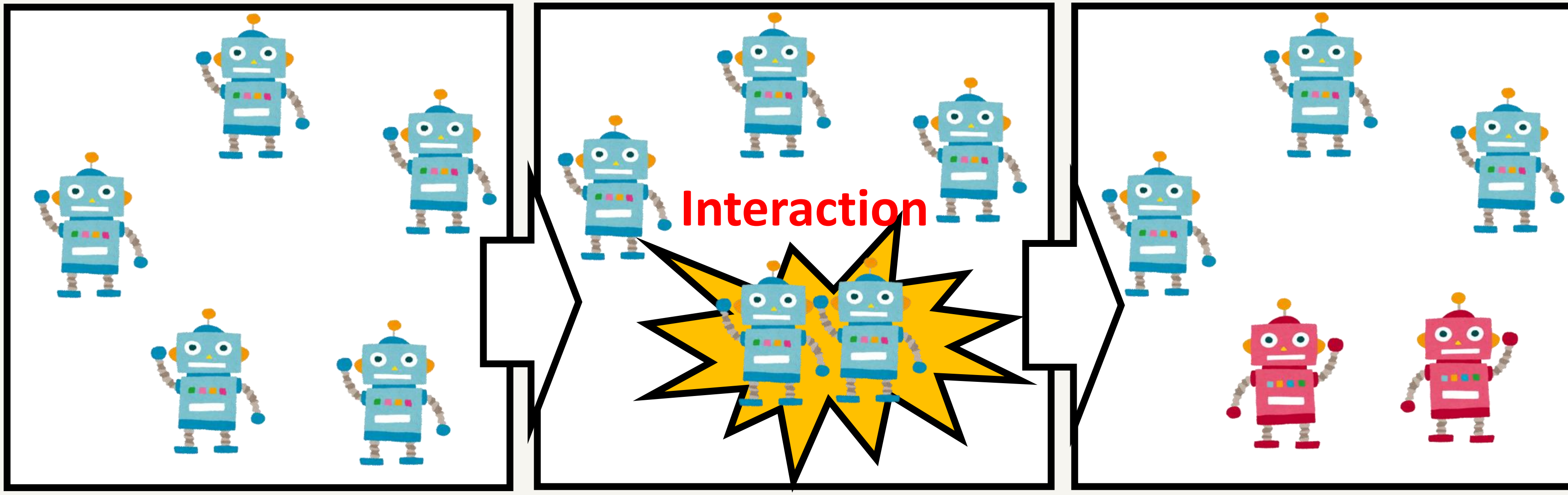
Hiroto Yasumi[1], Fukuhito Ooshita[2],
Ken'ichi Yamaguchi[1], and Michiko Inoue[2]
1 National Institute of Technology, Nara College
2 Nara Institute of Science and Technology

Outline

- Population Protocol
- Uniform bipartition Problem
- Results
  - Constant-space solvability for uniform bipartition
  - Space-optimal protocols for solvable cases
- Conclusion

Population protocol model [1]

- A model of multiple passively moving agents
  - Agents are uniform state machines
  - When two agents approach, update their states by interaction

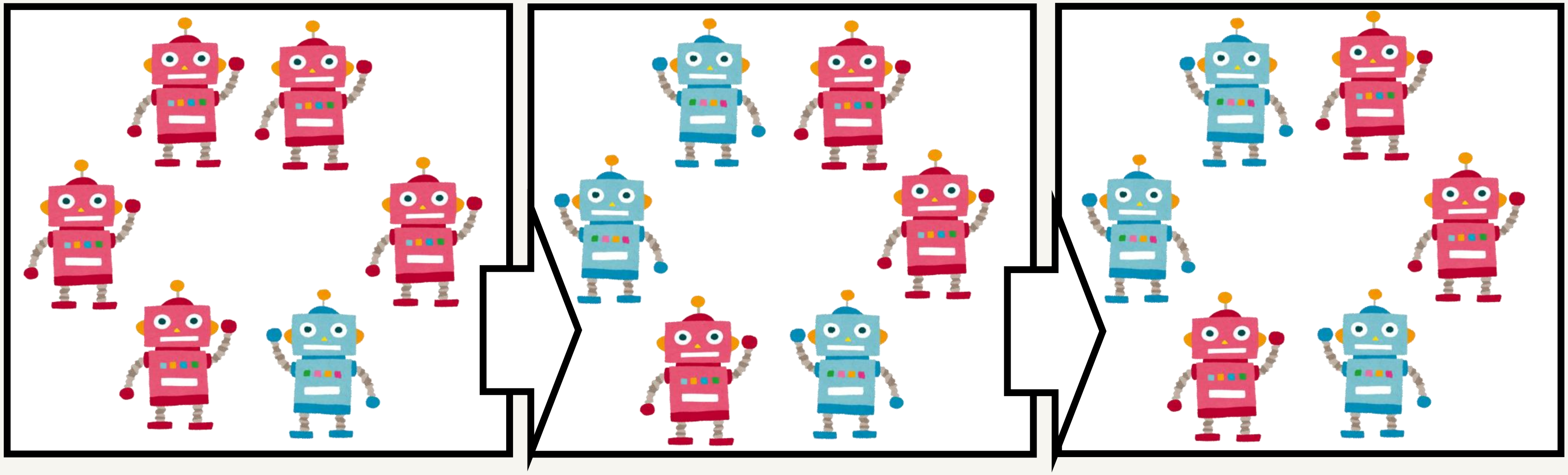


- Application example
  - Sensor networks to monitor wild birds
  - Molecular robot networks

[1] D. Angluin et al.,"Computation in networks of passively mobile finite-state sensors", Distributed Computing, Vol.18, No.4, pp.235-253(2006)

Uniform bipartition problem

- Divide a population into two groups of the same size
- Maintain the group after that



Divided into two groups

- Application example
  - Assign a different task to each group
  - Reduce energy consumption by switching on only one group

Related work [16]

- Uniform k-partition
  - Divide a population into k groups of the same size
- Some protocols are proposed
- Complexity is not studied

We study space complexity of k=2

[16] Carole Delporte-Gallet,et al., " When birds die: Making population protocols fault-tolerant", *Distributed Computing in Sensor Systems*, pages 51–66, (2006)

Result

- Clarify **constant-space solvability** for uniform bipartition
- Propose **space-optimal protocols** for solvable cases

The minimum number of states (n is the number of agents)

BS	Fairness	Designated initial states		Arbitrary initial states	
		Asymmetric	Symmetric	Asymmetric	Symmetric
Single	Global	3	3	4	4
	Weak	3	3	$\Omega(n)$	$\Omega(n)$
No	Global	3*	4*	Impossible	Impossible
	Weak	3*	Impossible	Impossible	Impossible

* Protocols are proposed in [16] and [14].

[16] C. Delporte-Gallet,et al., *Distributed Computing in Sensor Systems*, 2006
[14] O. Bournez, et al., In Proc. of the International Workshop on the Complexity of Simple Programs, 2008.

Result

- Propose protocols for six cases
- Clarify lower bounds for eleven cases
- Prove impossibility for five cases

The minimum number of states (n is the number of agents)

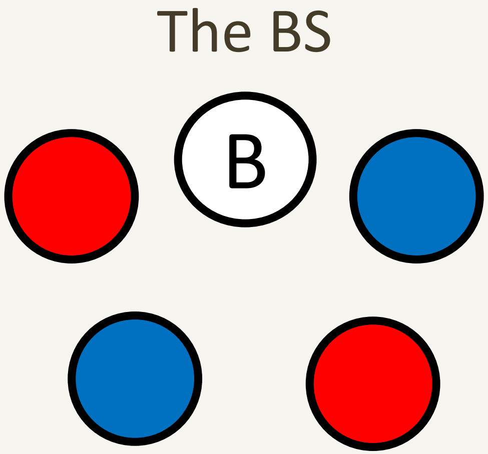
BS	Fairness	Designated initial states		Arbitrary initial states	
		Asymmetric	Symmetric	Asymmetric	Symmetric
Single	Global	3	3	4	4
	Weak	3	3	$\Omega(n)$	$\Omega(n)$
No	Global	3*	4*	Impossible	Impossible
	Weak	3*	Impossible	Impossible	Impossible

* Protocols are proposed in [16] and [14].

[16] C. Delporte-Gallet,et al., *Distributed Computing in Sensor Systems*, 2006
[14] O. Bournez, et al., In Proc. of the International Workshop on the Complexity of Simple Programs, 2008.

BS(base station)

- With a BS
  - A special agent (BS) exists
    - The BS is distinguishable
    - The BS has a powerful capability
      - We do not care the number of states of the BS
  - Other agents are identical
- Without a BS
  - All agents are identical

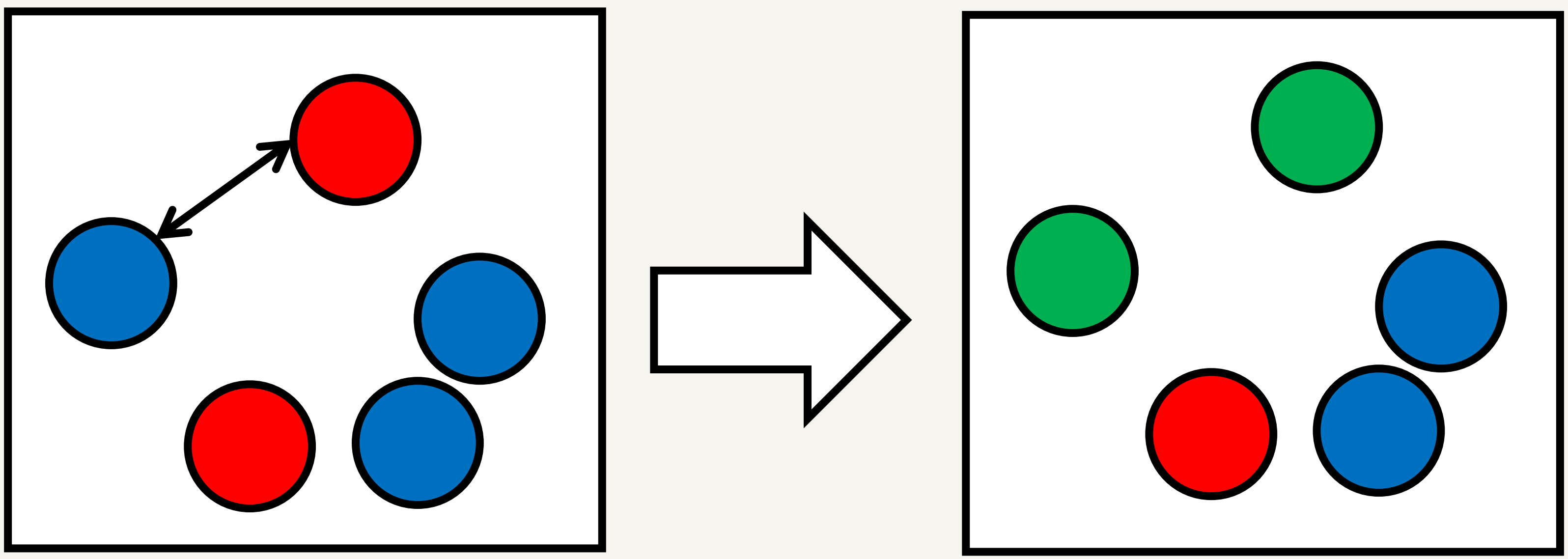


Fairness

- Assumption on interaction patterns of agents
- Types of fairness
  - Global fairness
  - Weak fairness

Configuration

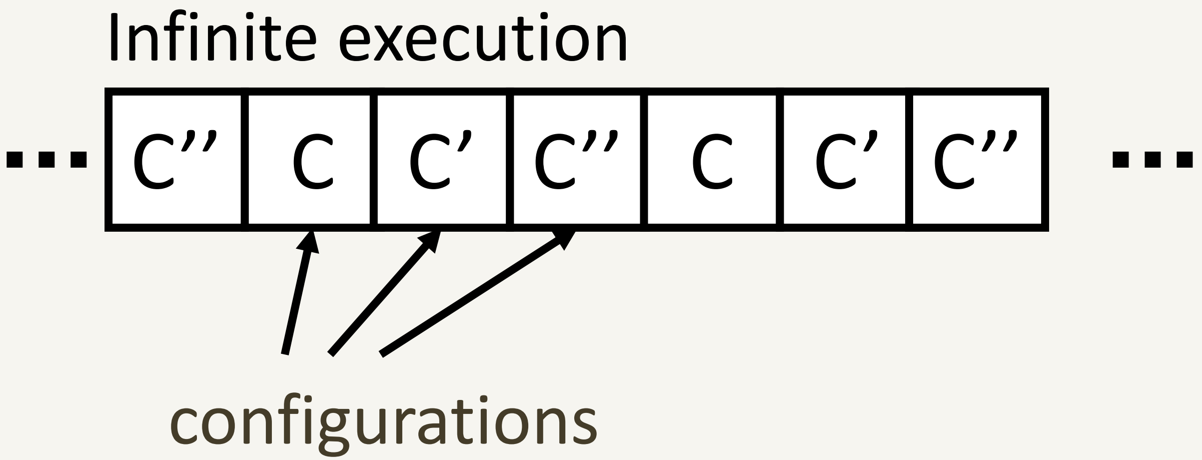
- A global state of a population
  - A combination of states of all agents
- If states transit, a configuration also transits
  - Any pair of agents can interact



configuration C configuration C’

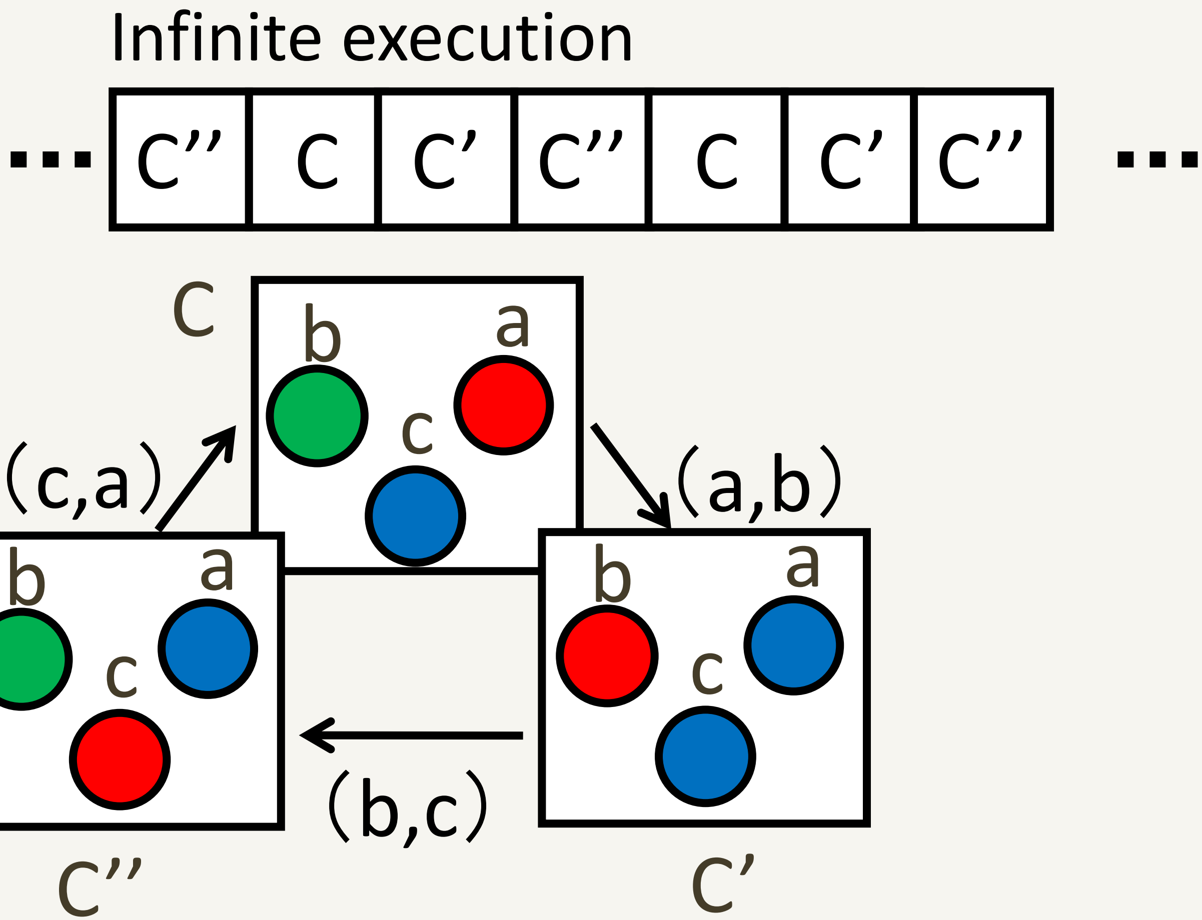
Global fairness

- If a configuration C occurs infinitely often, every configuration reachable from C occurs infinitely often



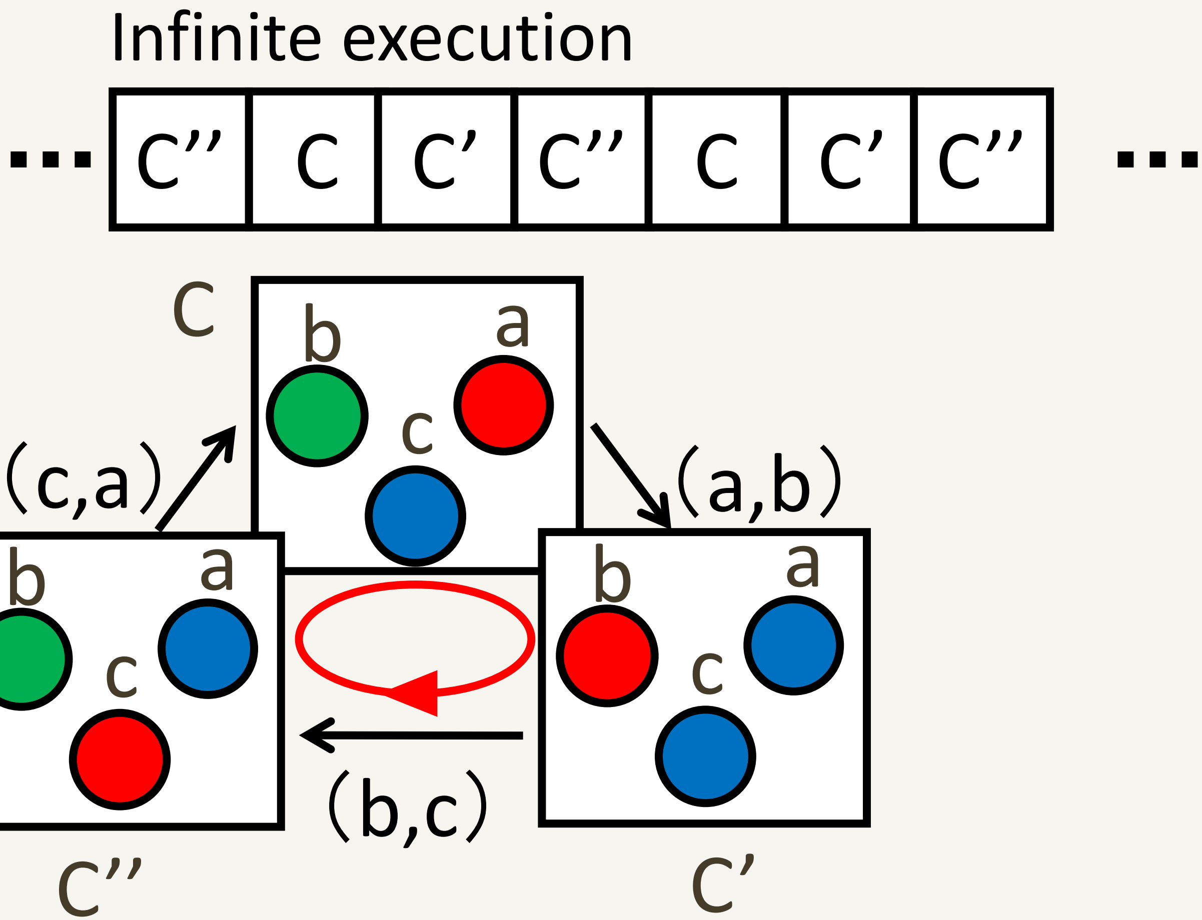
Global fairness

- If a configuration C occurs infinitely often, every configuration reachable from C occurs infinitely often



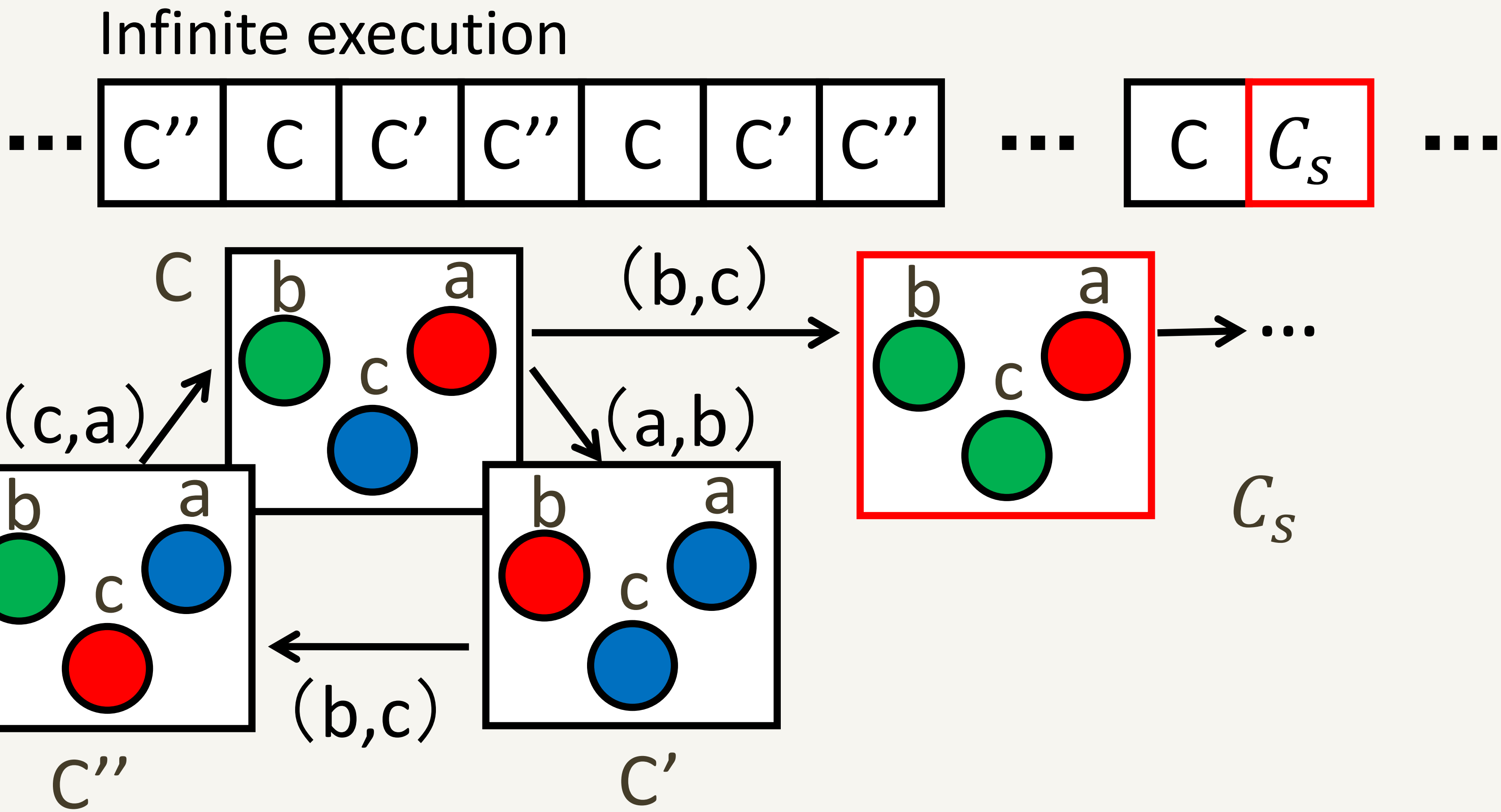
Global fairness

- If a configuration C occurs infinitely often, every configuration reachable from C occurs infinitely often



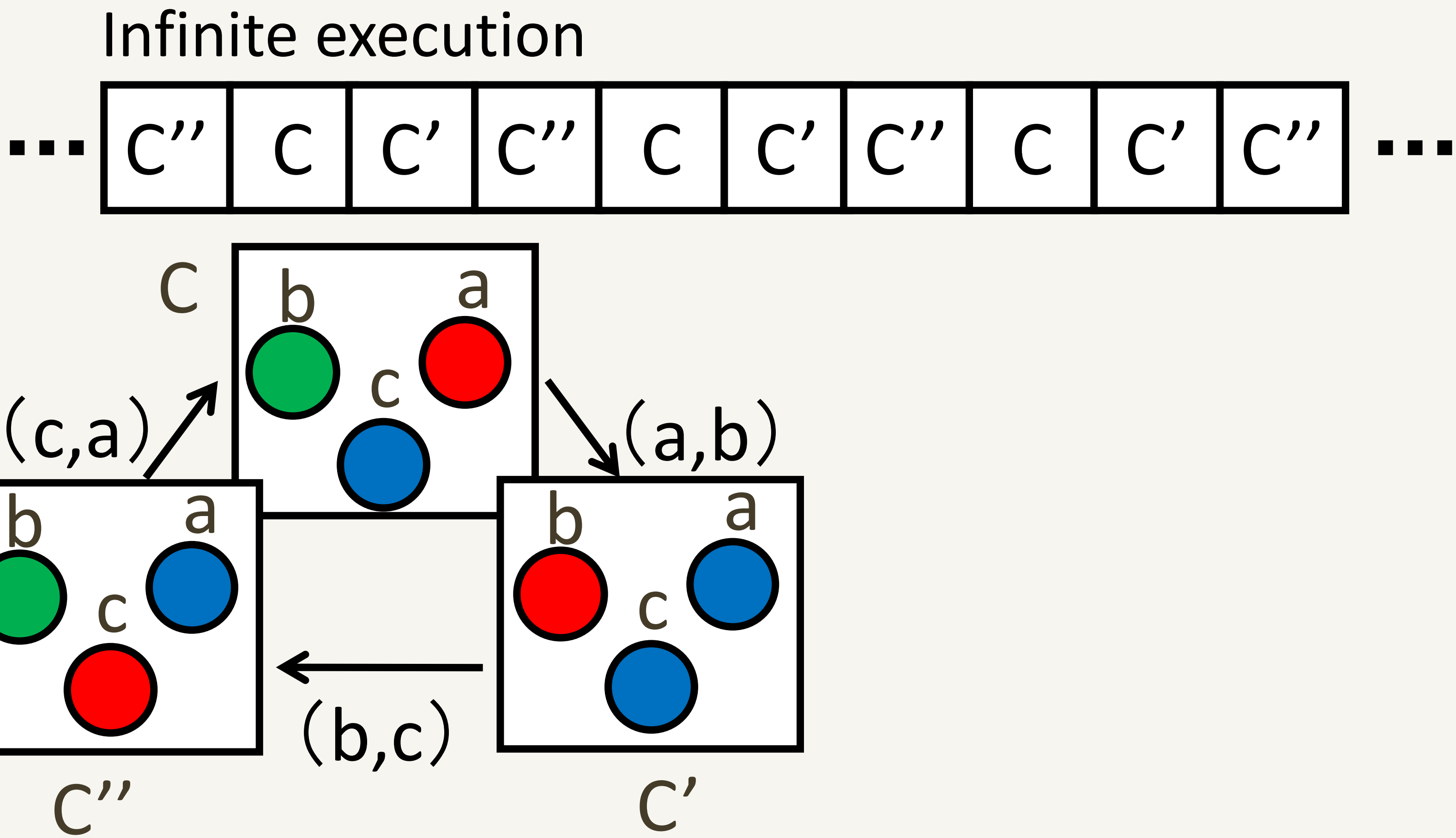
Global fairness

- If a configuration C occurs infinitely often, every configuration reachable from C occurs infinitely often



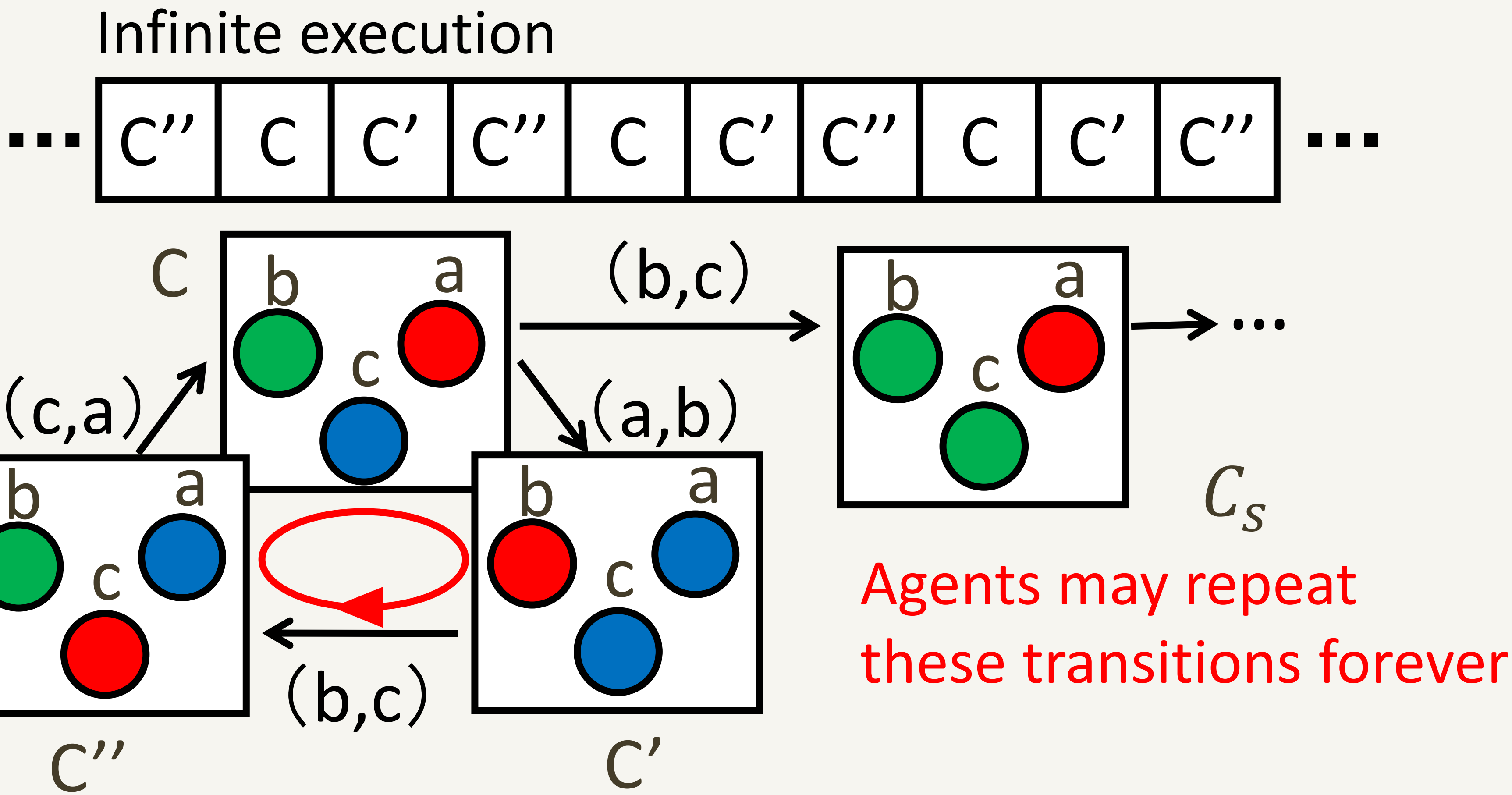
Weak fairness

- Interaction occurs infinitely often between each pair of agents



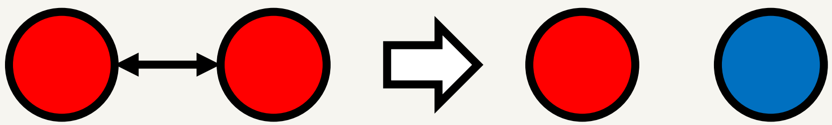
Weak fairness

- Interaction occurs infinitely often between each pair of agents

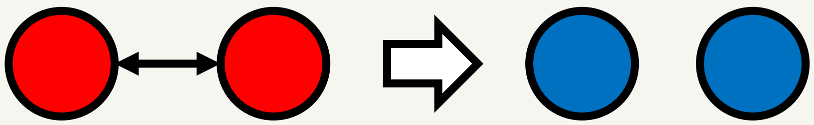


Symmetry

- Asymmetric protocol
  - Can include asymmetric transition
  - Asymmetric transition
    - Two agents with the same states transit to different states
- Symmetric protocol
  - Cannot include asymmetric transition



Asymmetric transition

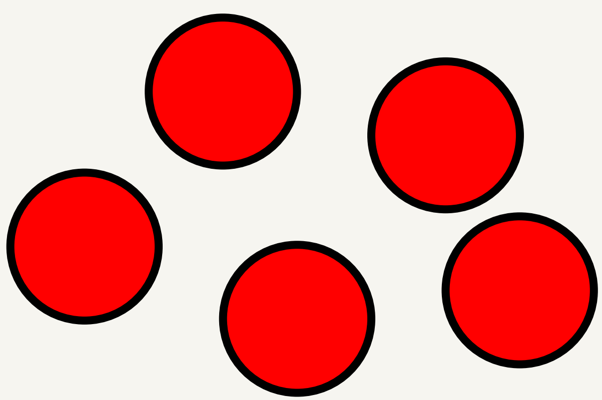


Not asymmetric transition

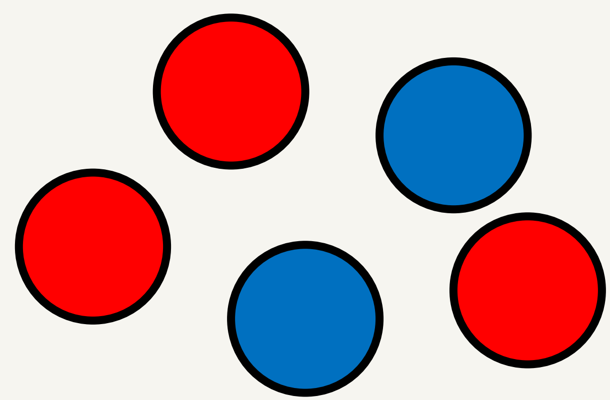
Requires a mechanism to break symmetry

Initial states

- Designated initial states
  - Need an initialization to use the protocol
- Arbitrary initial states
  - Do not need an initialization except for a BS
    - BS starts with a designated initial state
  - Tolerate transient faults



The designated initial state



The arbitrary initial state

Result

The minimum number of states (n is the number of agents)

BS	Fairness	Designated initial states		Arbitrary initial states	
		Asymmetric	Symmetric	Asymmetric	Symmetric
Single	Global	3	3	4	4
	Weak	3	3	$\Omega(n)$	$\Omega(n)$
No	Global	3*	4*	Impossible	Impossible
	Weak	3*	Impossible	Impossible	Impossible

* Protocols are proposed in [16] and [14].

[16] C. Delporte-Gallet,et al., *Distributed Computing in Sensor Systems*, 2006
[14] O. Bournez, et al., In Proc. of the International Workshop on the Complexity of Simple Programs, 2008.

Result

The minimum number of states (n is the number of agents)

BS	Fairness	Designated initial states		Arbitrary initial states	
		Asymmetric	Symmetric	Asymmetric	Symmetric
Single	Global	3	3	4	4
	Weak	3	3	$\Omega(n)$	$\Omega(n)$
No	Global	3*	4*	Impossible	Impossible
	Weak	3*	Impossible	Impossible	Impossible

* Protocols are proposed in [16] and [14].

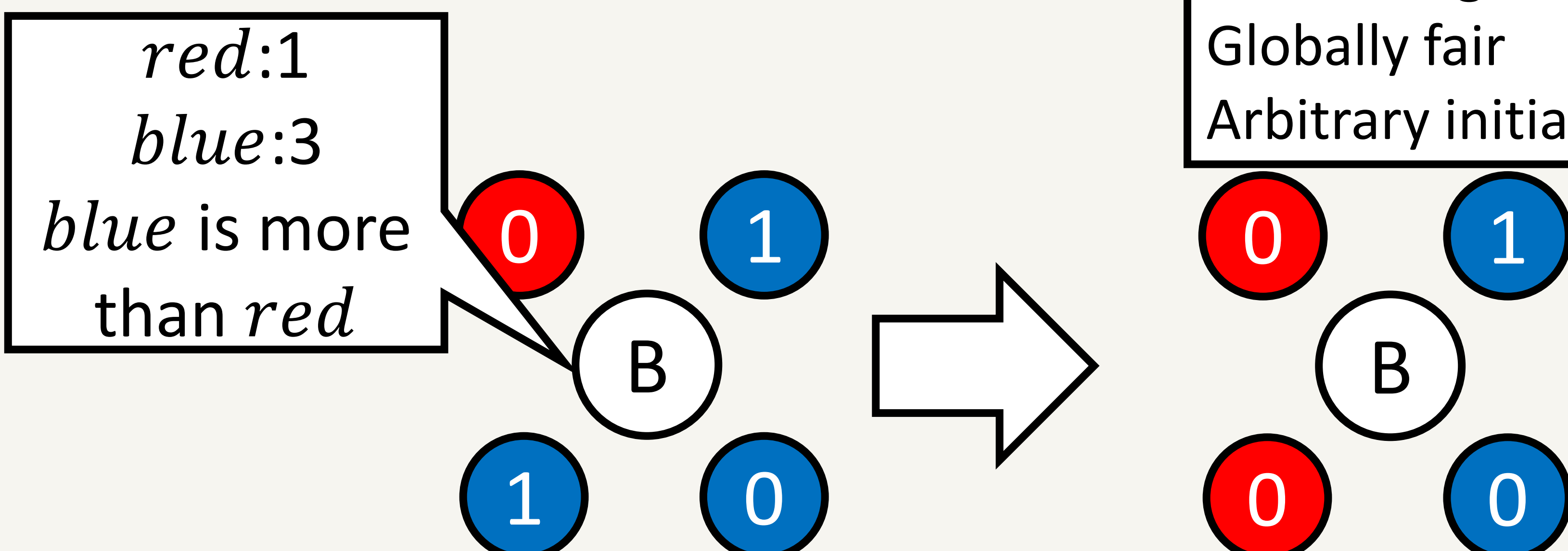
[16] C. Delporte-Gallet,et al., *Distributed Computing in Sensor Systems*, 2006
[14] O. Bournez, et al., In Proc. of the International Workshop on the Complexity of Simple Programs, 2008.

Overview of our protocol

- Use four states 

- Count the number of *red* agents and *blue* agents
  - Use counting protocol
    - Two states (0/1) for each agent
- Whenever the balance of *red* and *blue* is not maintained, adjust the number of agents

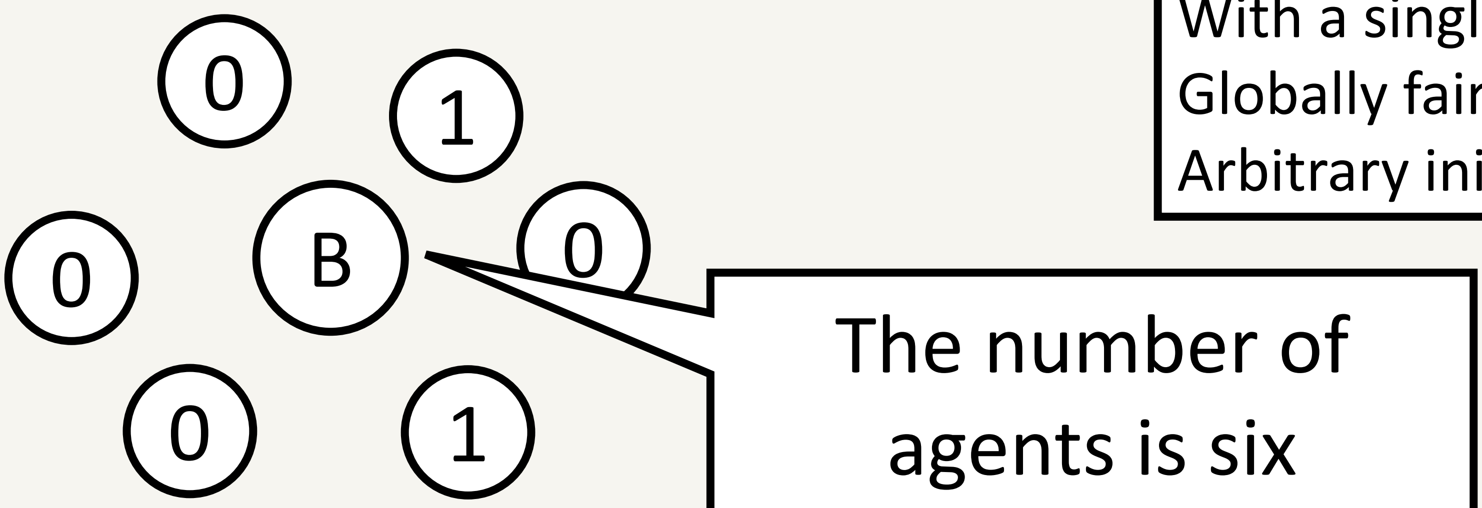
With a single BS
Globally fair
Arbitrary initial states



Counting protocol [9]

- The BS counts the number of agents
  - The count starts from 0 and eventually converges to the number of agents
- Each agent uses only two states
- The assumption is the same as our protocol

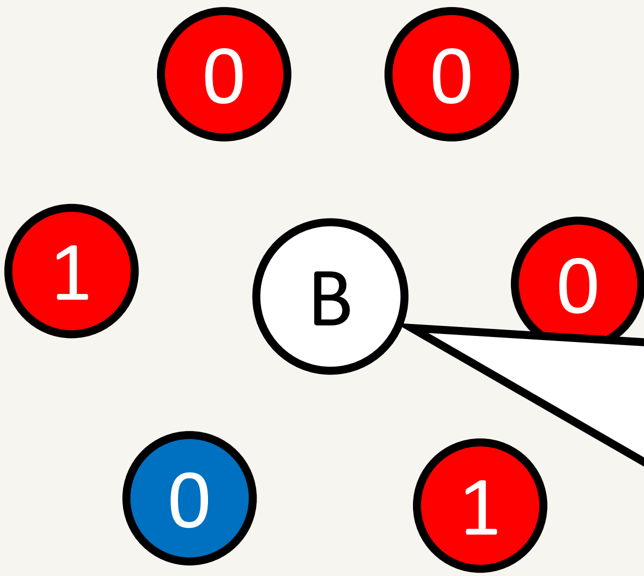
With a single BS
Globally fair
Arbitrary initial states



[9] J. Aspnes et al., "Time and space optimal counting in population protocols" ,*OPODIS2016*.

Usage of counting protocol

- The BS counts the numbers of *red* and *blue* agents using two states 0/1

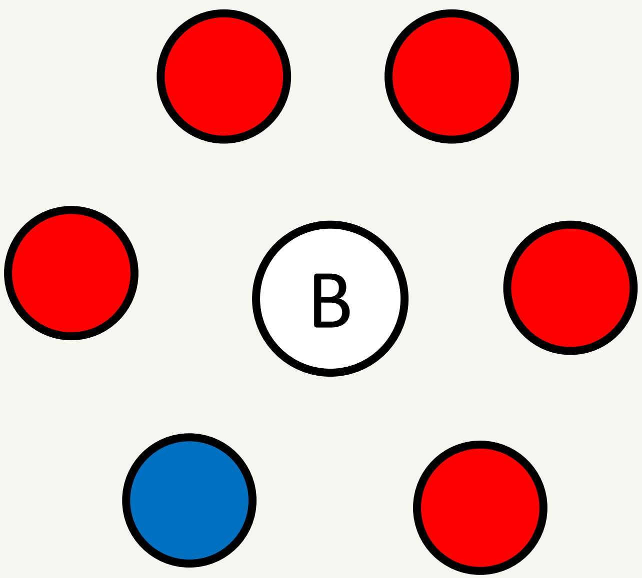


The BS in parallel executes two instances of counting protocol

- $Count_{red}$
  - Output # of *red* agents
- $Count_{blue}$
  - Output # of *blue* agents

How to keep # of $red/blue$ agents

- BS keeps variables red and $blue$
  - $red = r$ implies BS knows the current number of red agents is at least r
  - $blue = b$ implies BS knows the current number of blue agents is at least b



red:0

$blue$:0

$Count_{red}$ output:0

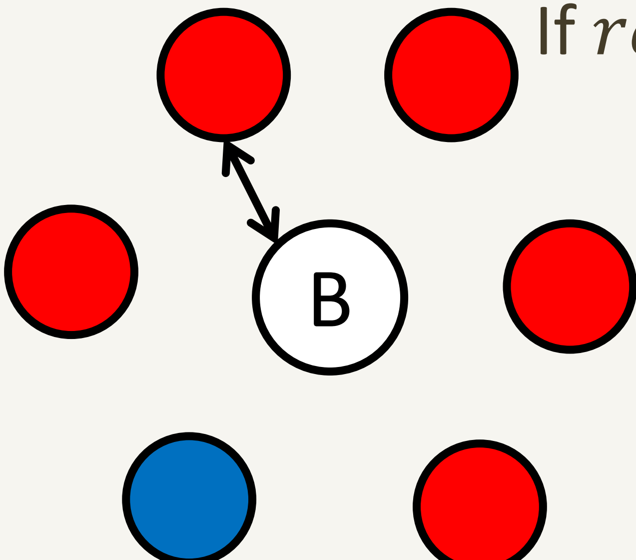
$Count_{blue}$ output:0

With a single BS
Globally fair
Arbitrary initial states

How to keep # of $red/blue$ agents

- BS keeps variables red and $blue$
  - $red = r$ implies
    BS knows the current number of red agents is at least r
  - $blue = b$ implies
    BS knows the current number of blue agents is at least b

If $red < Count_{red}$, assign red to $Count_{\mathrm{red}}$



red:0

$blue$:0

$Count_{red}$ output:1

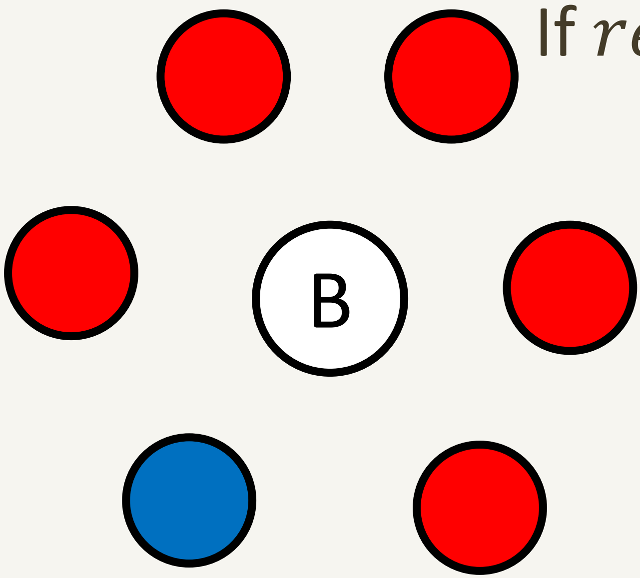
$Count_{blue}$ output:0

With a single BS
Globally fair
Arbitrary initial states

How to keep # of $red/blue$ agents

- BS keeps variables red and $blue$
  - $red = r$ implies
    BS knows the current number of red agents is at least r
  - $blue = b$ implies
    BS knows the current number of blue agents is at least b

If $red < Count_{red}$, assign red to $Count_{\mathrm{red}}$



red:1

$blue$:0

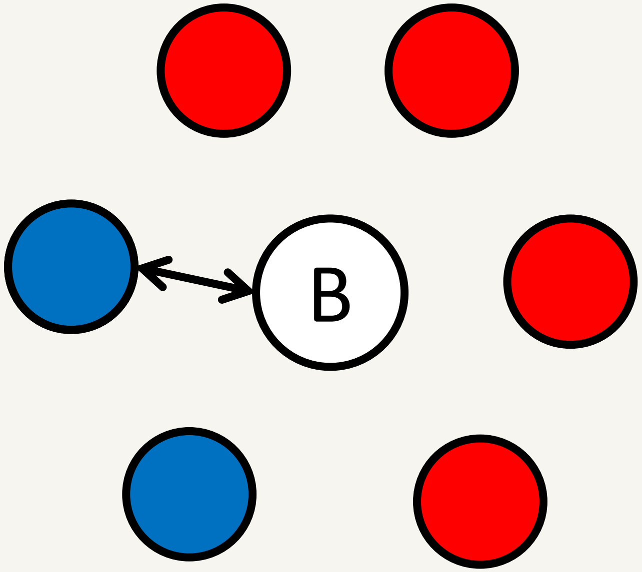
$Count_{red}$ output:1

$Count_{blue}$ output:0

With a single BS
Globally fair
Arbitrary initial states

How to adjust # of $red/blue$ agents

- Execute the following if $red \geq blue + 2$
  1. Change a red agent to $blue$
  2. Update variables red and $blue$
  3. Reset counting protocols



red: 1

$blue$: 1

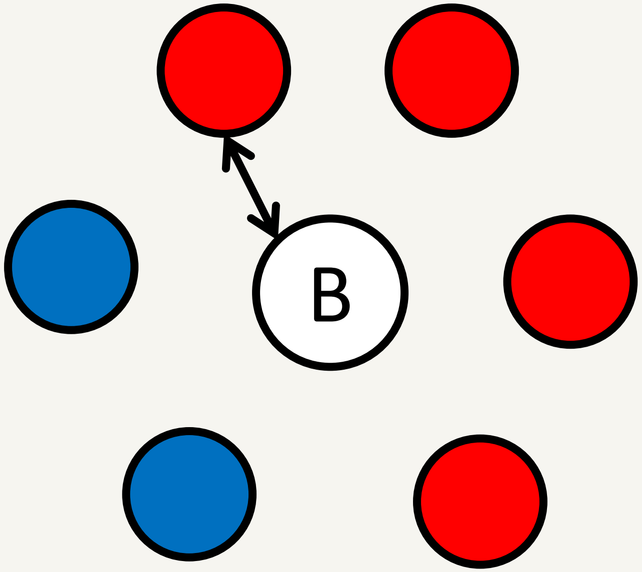
$Count_{red}$ output: 0

With a single BS
Globally fair
Arbitrary initial states

Execute similarly if $blue \geq red + 2$

Convergence to uniform bipartition

- Eventually variables red and $blue$ become the same
  → Uniform bipartition is achieved



red:1

$blue$:1

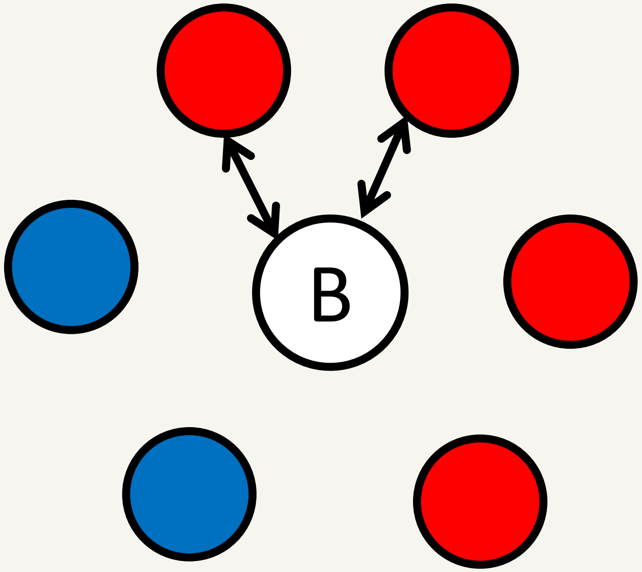
$Count_{red}$ output:1

$Count_{blue}$ output:0

With a single BS
Globally fair
Arbitrary initial states

Convergence to uniform bipartition

- Eventually variables red and $blue$ become the same
  → Uniform bipartition is achieved

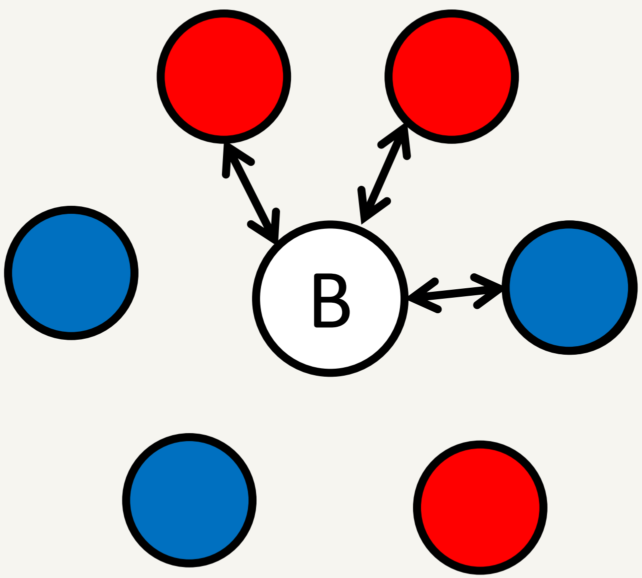


red:2
$blue$:1
$Count_{red}$ output:2
$Count_{blue}$ output:0

With a single BS
Globally fair
Arbitrary initial states

Convergence to uniform bipartition

- Eventually variables red and $blue$ become the same
  → Uniform bipartition is achieved



red:3

$blue$:1

$Count_{red}$ output:3

$Count_{blue}$ output:0

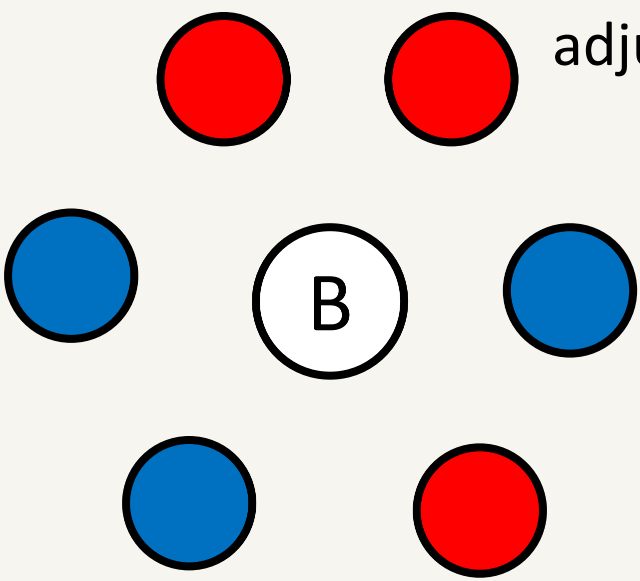
With a single BS
Globally fair
Arbitrary initial states

Convergence to uniform bipartition

- Eventually variables red and $blue$ become the same
  → Uniform bipartition is achieved

Whenever the difference becomes two or more, adjust # of red and $blue$ agents

With a single BS
Globally fair
Arbitrary initial states



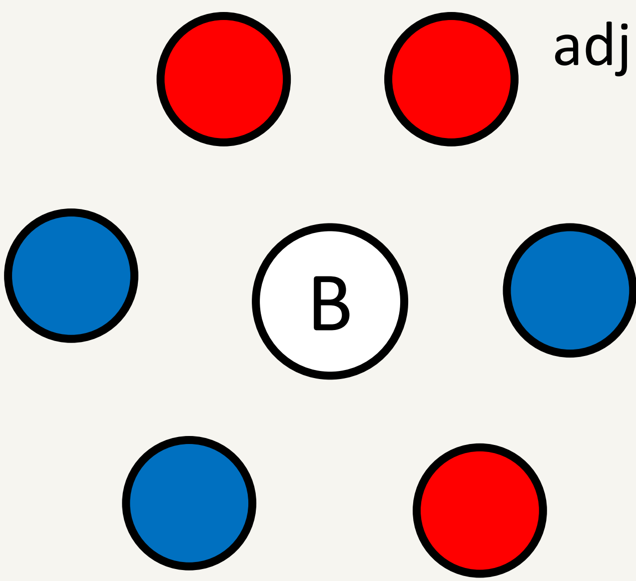
red:2
$blue$:2
$Count_{red}$ output:0
$Count_{blue}$ output:0

Convergence to uniform bipartition

- Eventually variables red and $blue$ become the same
  → Uniform bipartition is achieved

Whenever the difference becomes two or more, adjust # of red and $blue$ agents

With a single BS
Globally fair
Arbitrary initial states



red:3
$blue$:3
$Count_{red}$ output:3
$Count_{blue}$ output:3

Result

The minimum number of states (n is the number of agents)

BS	Fairness	Designated initial states		Arbitrary initial states	
		Asymmetric	Symmetric	Asymmetric	Symmetric
Single	Global	3	3	4	4
	Weak	3	3	$\Omega(n)$	$\Omega(n)$
No	Global	3*	4*	Impossible	Impossible
	Weak	3*	Impossible	Impossible	Impossible

* Protocols are proposed in [16] and [14].

[16] C. Delporte-Gallet,et al., *Distributed Computing in Sensor Systems*, 2006

[14] O. Bournez, et al., In Proc. of the International Workshop on the Complexity of Simple Programs, 2008.

Impossibility with three states

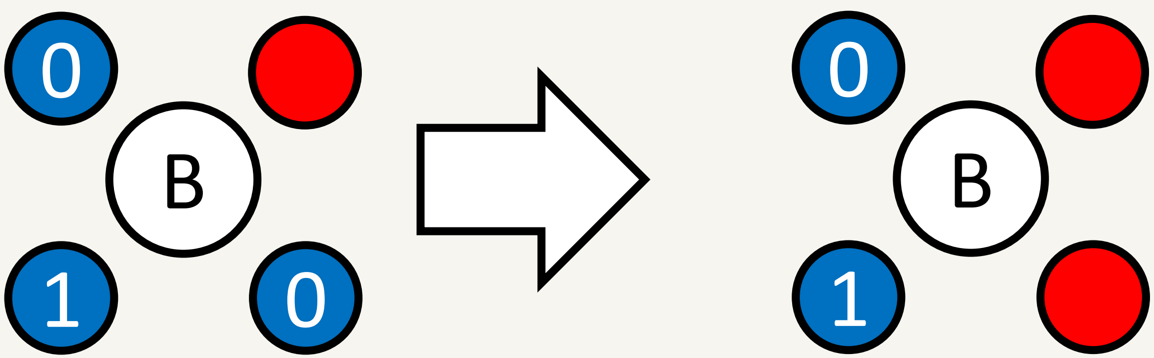
- For contradiction, assume a protocol with three states under the following assumptions

> With a single BS
> Globally fair
> Arbitrary initial states

- The protocol divides agents into red and blue agents
  - The numbers of red and blue agents are equal
- W.l.o.g., we assume two states for blue agents 0 1 and one state for red agents

Impossibility with three states

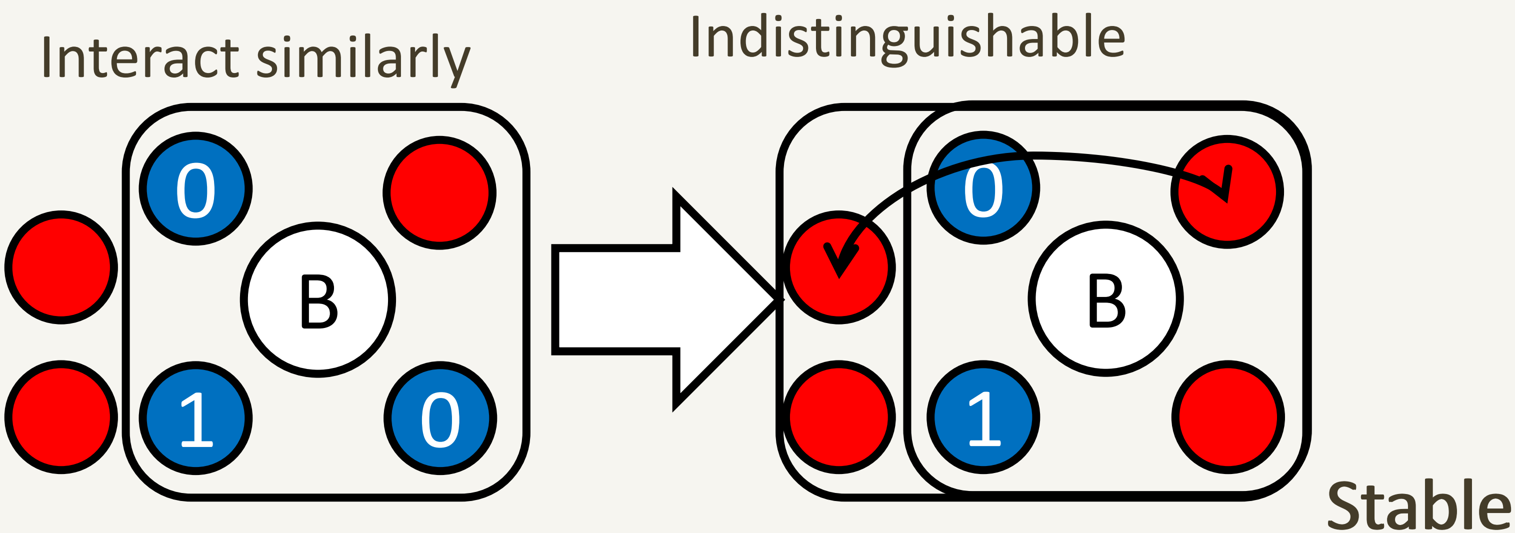
1. Case of n agents
   - The protocol divides population into $\frac{n}{2}$ red/blue agents



After stabilizing, red agents never transit to blue

Impossibility with three states

2. Case of $n+2$ agents
   - n agents stabilize similarly to case of n agents
   - Red agents never change their groups after that



After stabilizing, red agent never transit to blue

Conclusion

- Clarify constant-space solvability for uniform bipartition
- Propose space-optimal protocols for solvable cases

The minimum number of states (n is the number of agents)

BS	Fairness	Designated initial states		Arbitrary initial states	
		Asymmetric	Symmetric	Asymmetric	Symmetric
Single	Global	3	3	4	4
	Weak	3	3	$\Omega(n)$	$\Omega(n)$
No	Global	3*	4*	Impossible	Impossible
	Weak	3*	Impossible	Impossible	Impossible

* Protocols are proposed in [16] and [14].

[16] C. Delporte-Gallet,et al., *Distributed Computing in Sensor Systems*, 2006
[14] O. Bournez, et al., In Proc. of the International Workshop on the Complexity of Simple Programs, 2008.