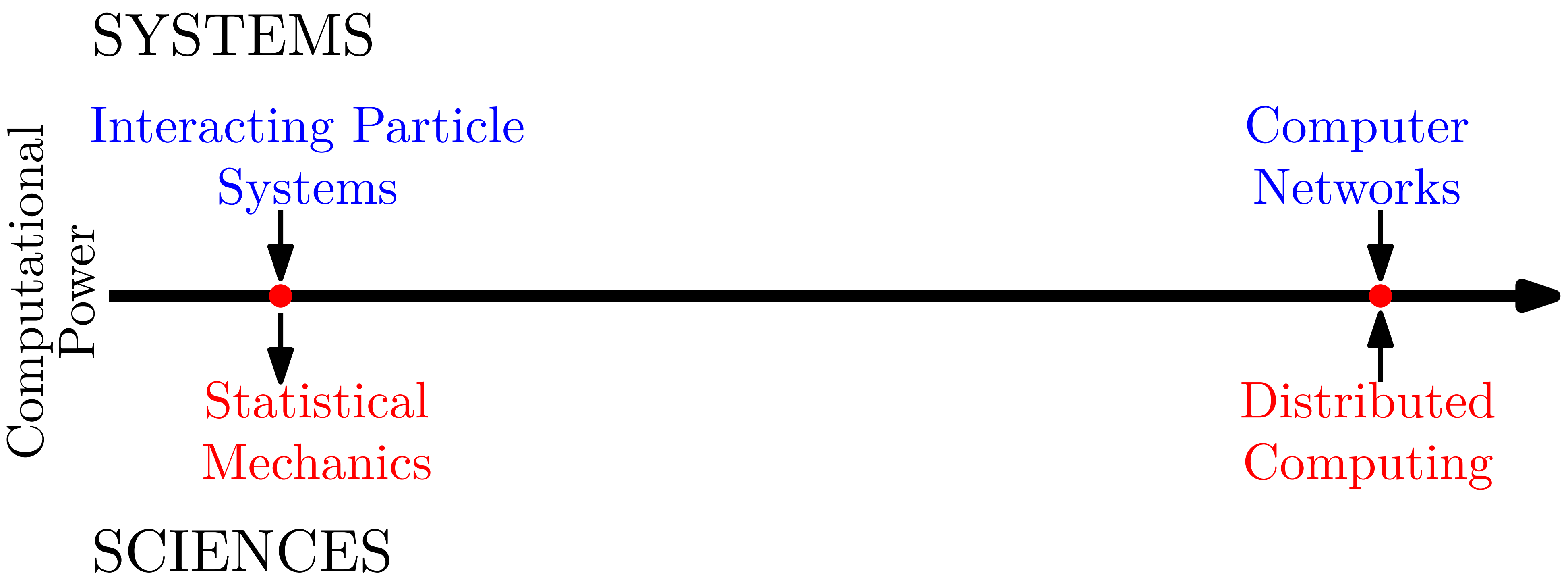
What can be Computed in a Simple Chaotic Way?

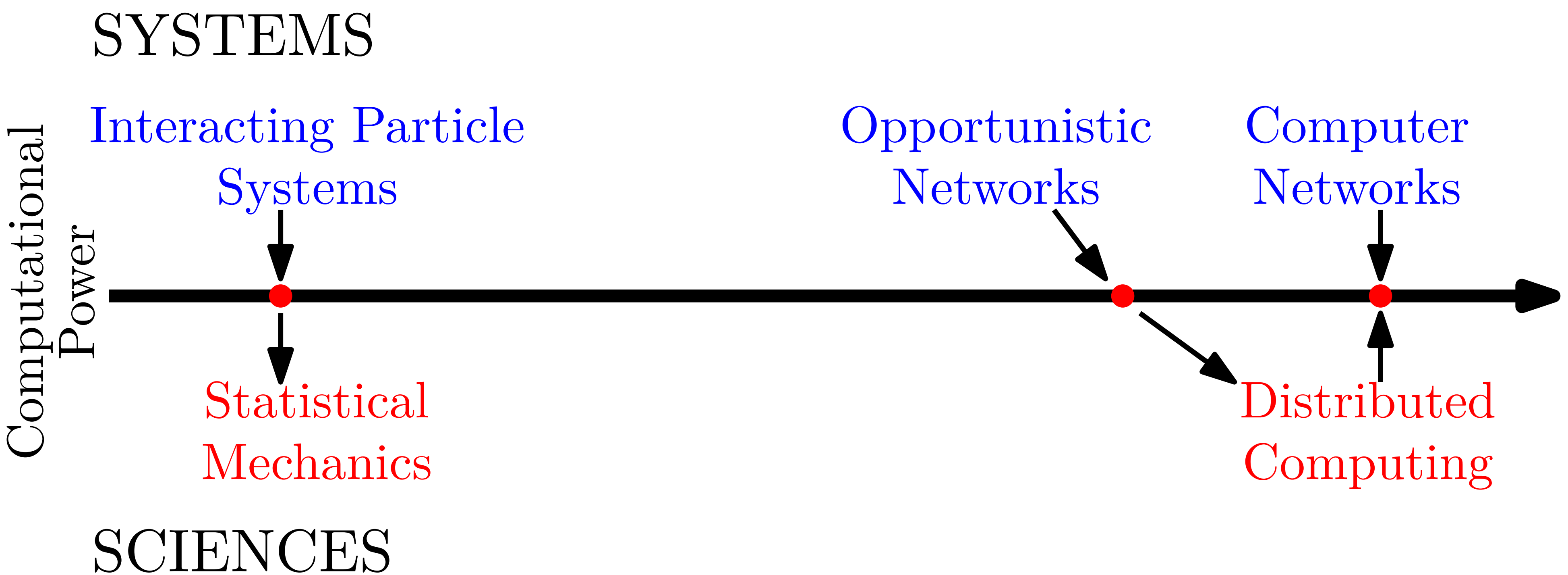
Emanuele Natale

max planck institut
informatik

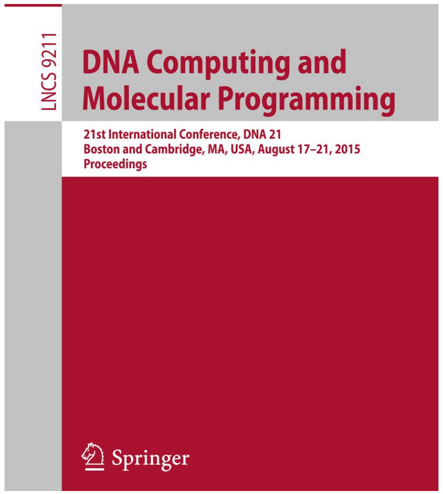
What can *Simple* Systems do?

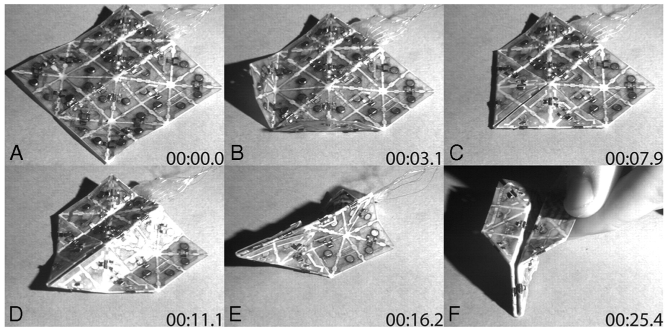


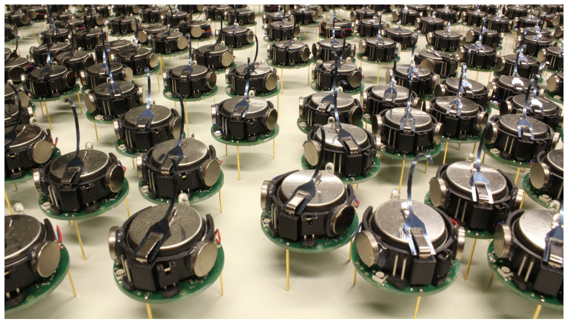
What can *Simple* Systems do?



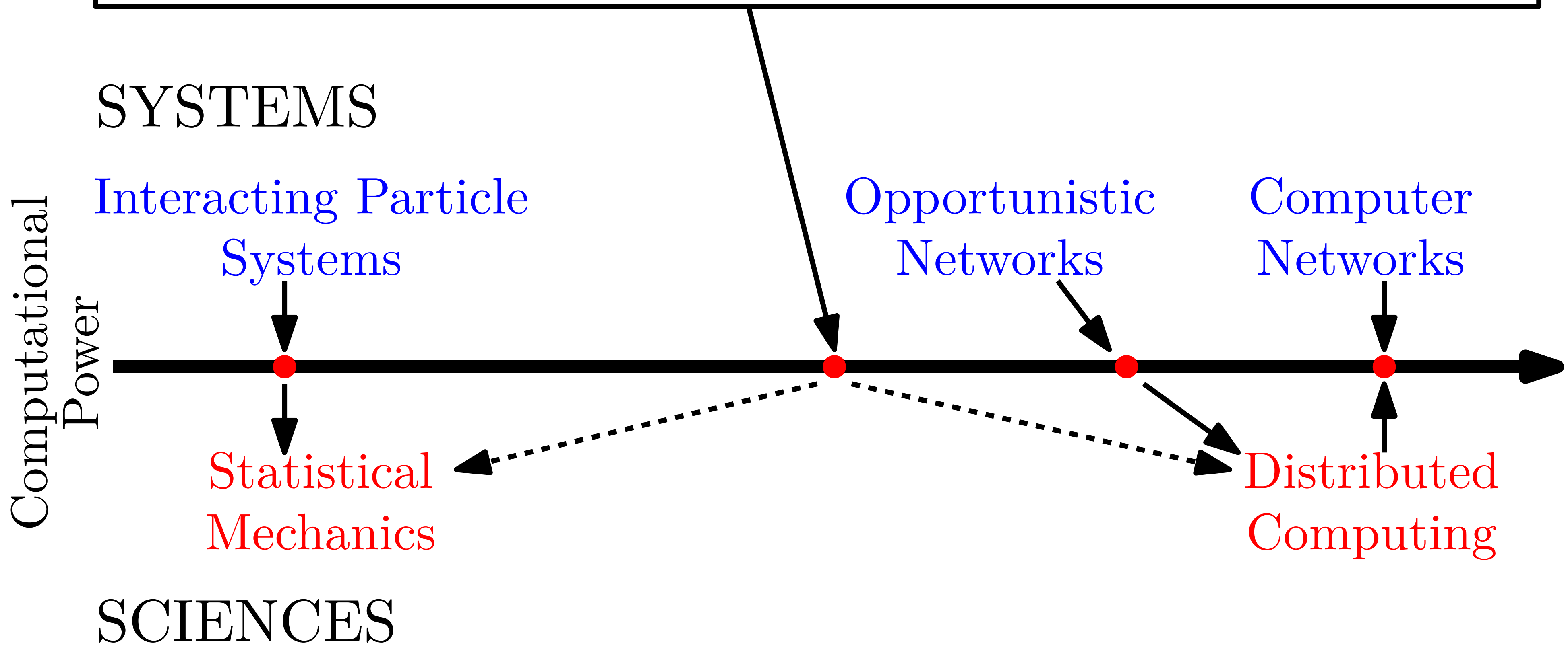
What can *Simple* Systems do?



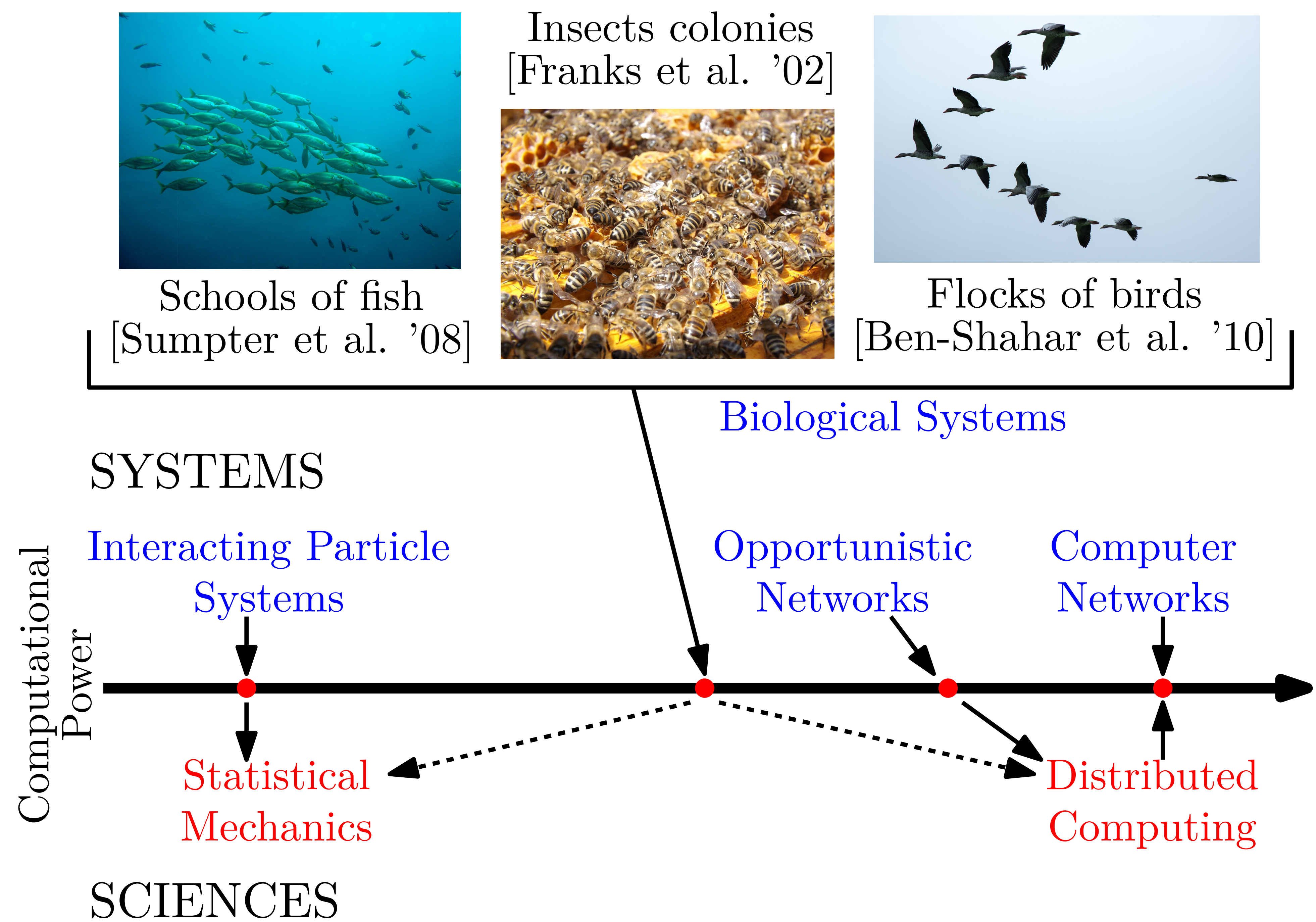




DNA/Molecular Computing, Programmable Matter, Swarms of Simple Robots



What can *Simple* Systems do?



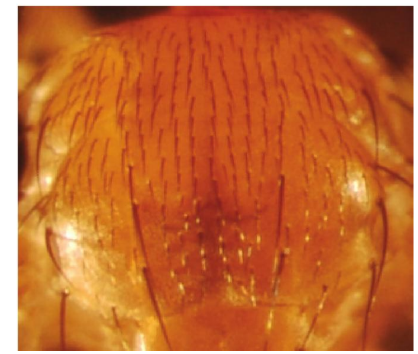
Examples of Natural Algorithms

How birds of flocks synchronize their flight [Chazelle '09]

Examples of Natural Algorithms

How birds of flocks synchronize their flight [Chazelle '09]

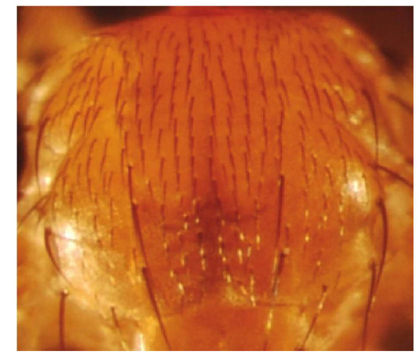
How are sensory organ precursor cells selected in a fly's nervous system [AABHBB '11]

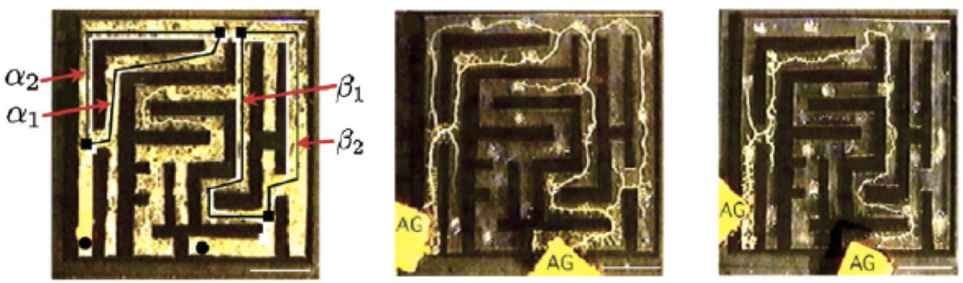


Examples of Natural Algorithms

How birds of flocks synchronize their flight [Chazelle '09]

How are sensory organ precursor cells selected in a fly's nervous system [AABHBB '11]

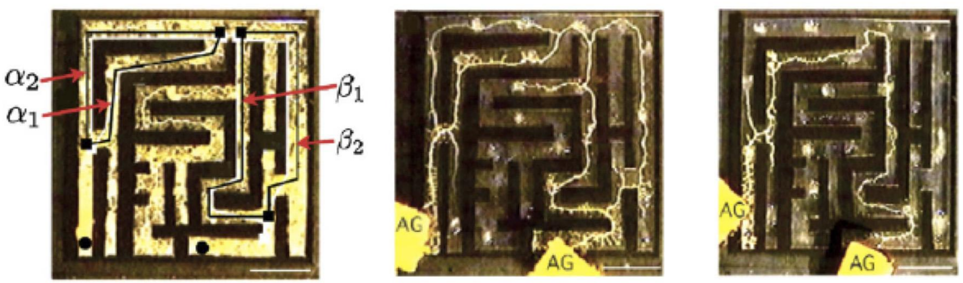




How Physarum polycephalum finds shortest paths [BBDKM '14]

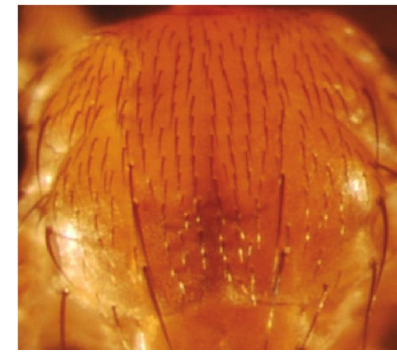
Examples of Natural Algorithms

How birds of flocks synchronize their flight [Chazelle '09]

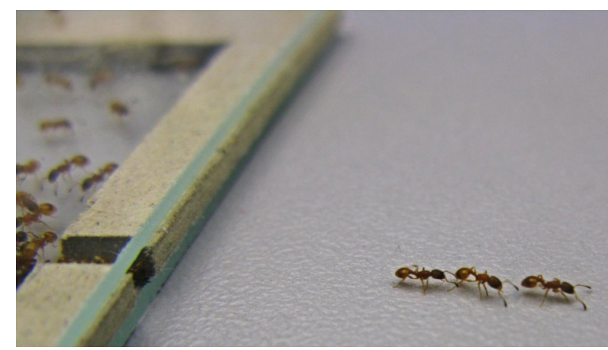


How Physarum polycephalum finds shortest paths [BBDKM '14]

How are sensory organ precursor cells selected in a fly's nervous system [AABHBB '11]



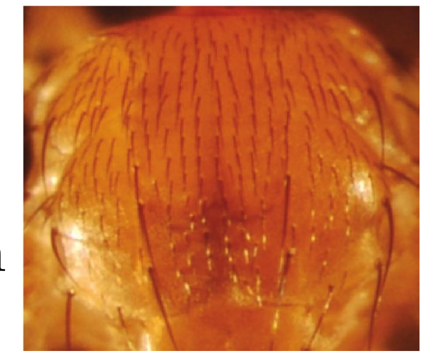
How do ants decide where to relocate their nest? [GMRL '15]

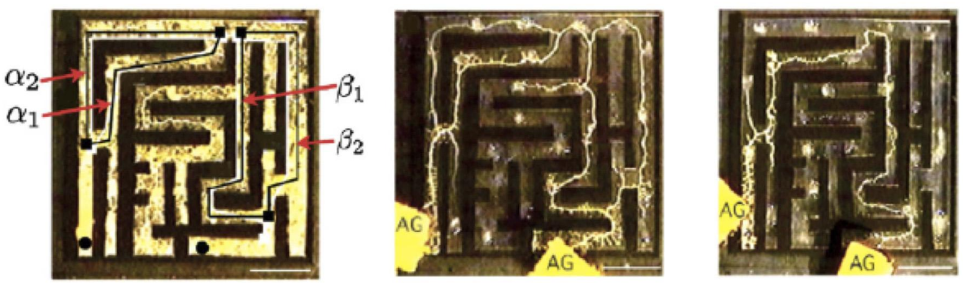


Examples of Natural Algorithms

How birds of flocks synchronize their flight [Chazelle '09]

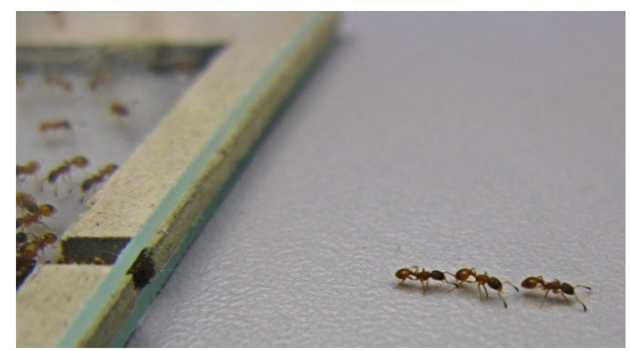
How are sensory organ precursor cells selected in a fly's nervous system [AABHBB '11]





How Physarum polycephalum finds shortest paths [BBDKM '14]

How do ants decide where to relocate their nest? [GMRL '15]



How ants perform collective navigattion [FHBGKKF '16]

Unstructured Communication Models

Animal communication:

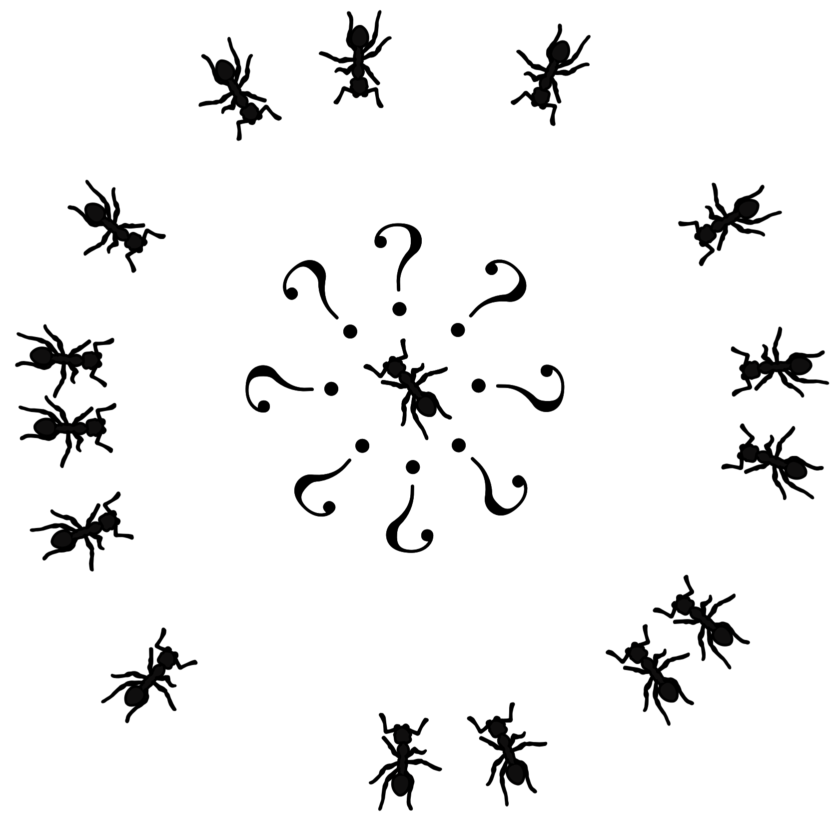
- Chaotic
- Anonymous
- Parsimonious
- Uni-directional (Passive/Active)
- Noisy

Unstructured Communication Models

Animal communication:

- Chaotic
- Anonymous
- Parsimonious
- Uni-directional (Passive/Active)
- Noisy

$\color{blue}{\mathcal{PULL}(h,\ell)}$ **model [Demers '88]**: at each round each agent can *observe* h other agents chosen independently and uniformly at random, and *shows* ℓ *bits* to her observers.

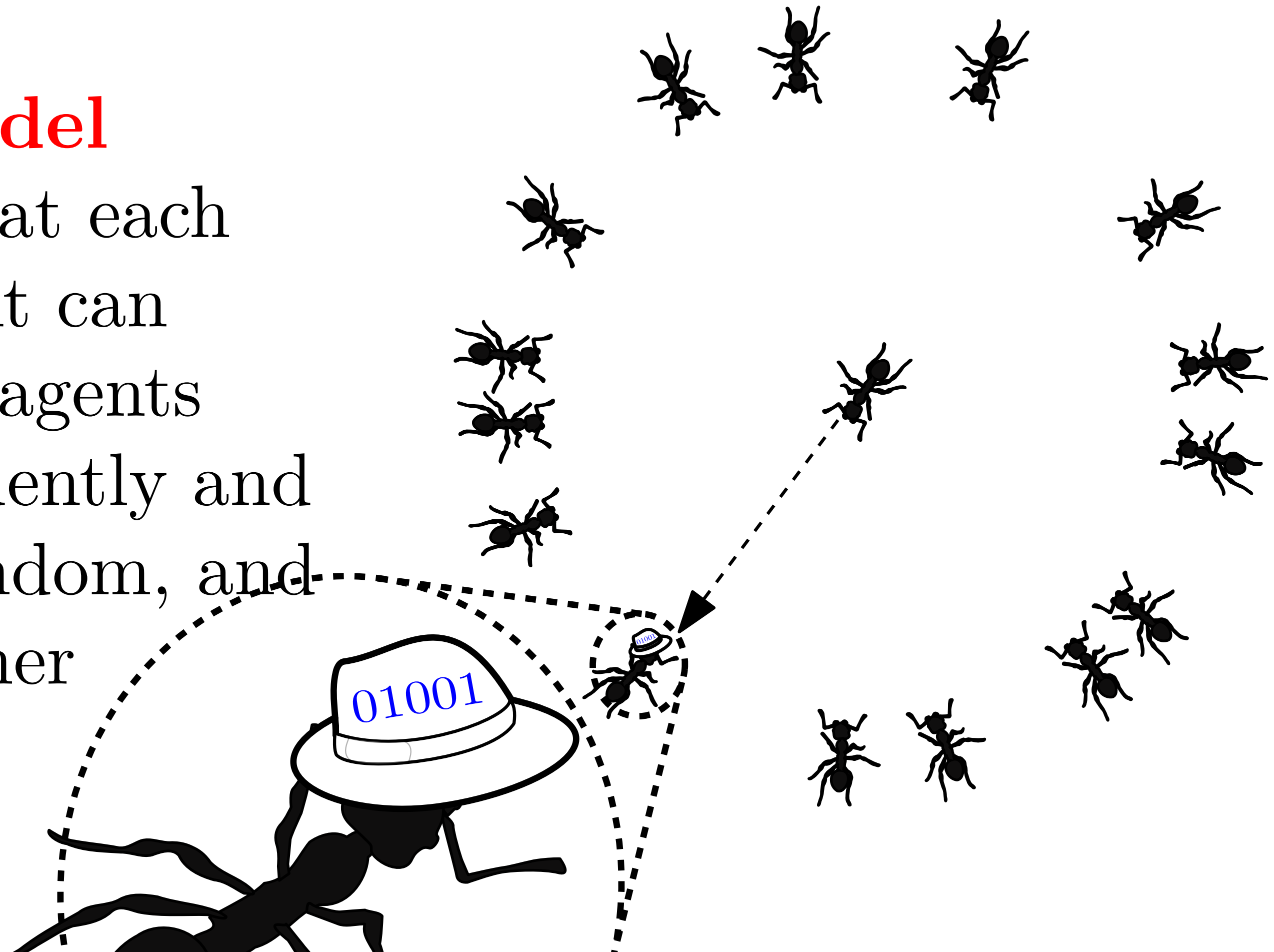


Unstructured Communication Models

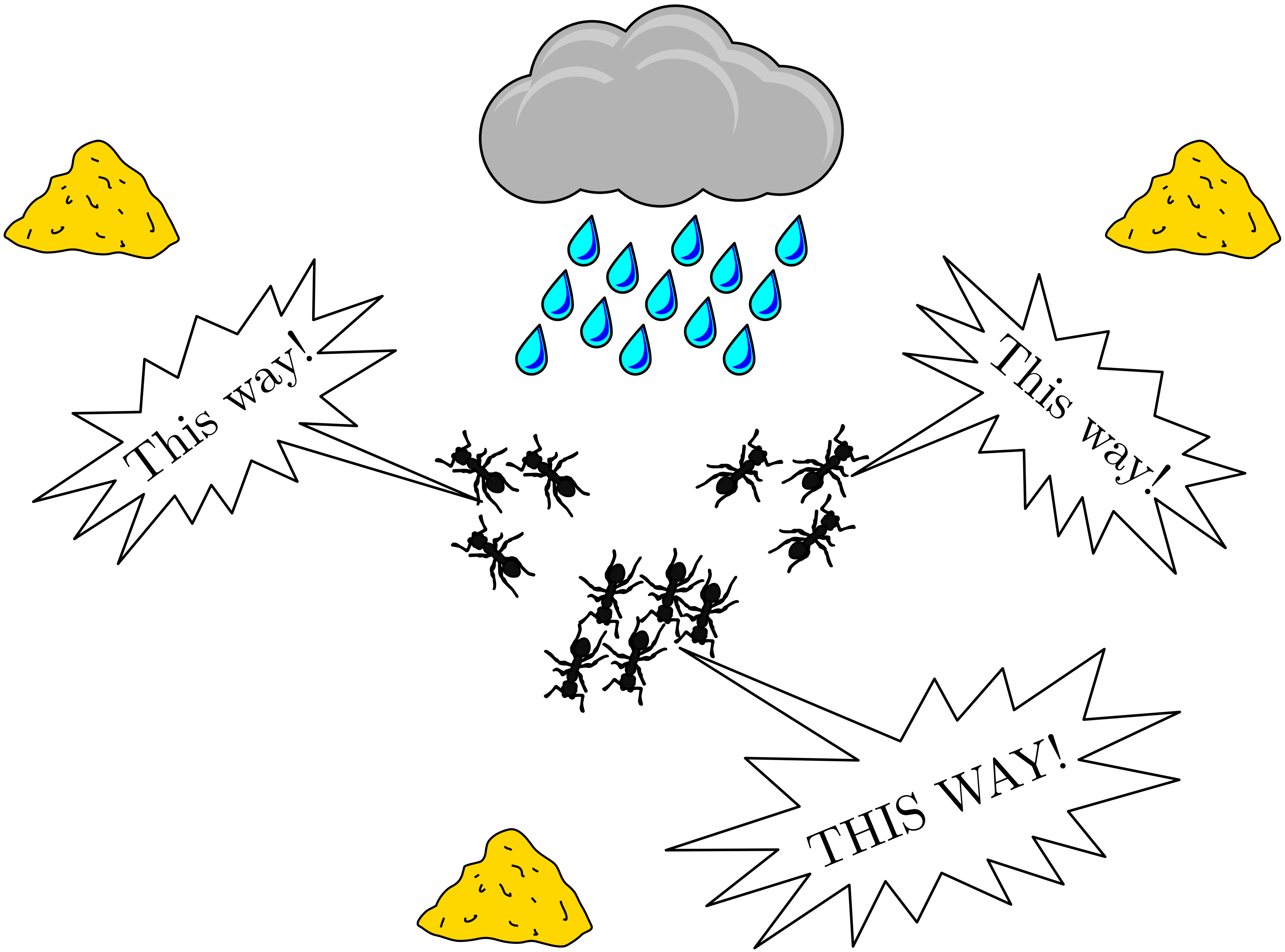
Animal communication:

- Chaotic
- Anonymous
- Parsimonious
- Uni-directional (Passive/Active)
- Noisy

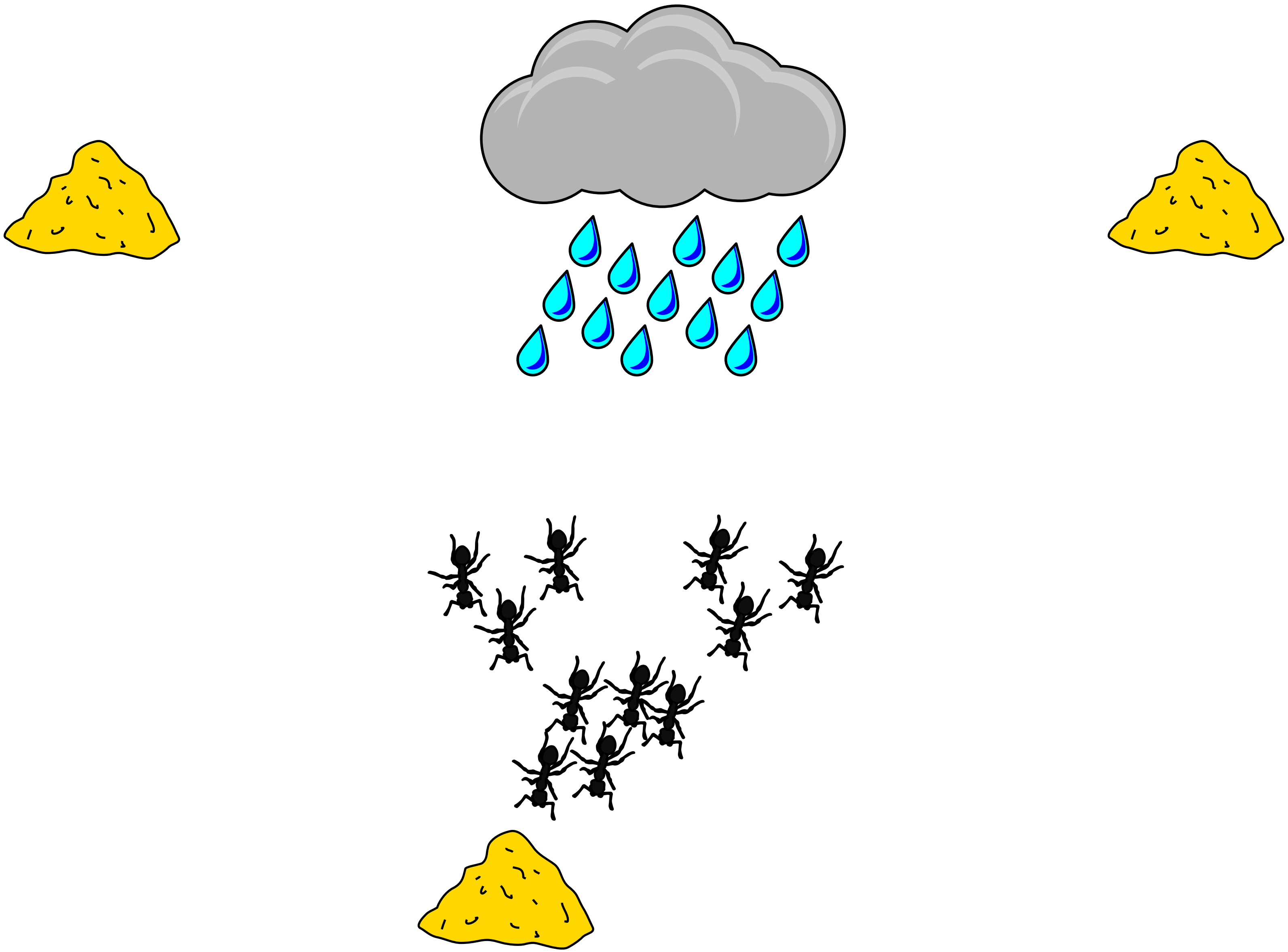
$\mathcal{PULL}(h,\ell)$ **model [Demers '88]**: at each round each agent can *observe* h other agents chosen independently and uniformly at random, and *shows* ℓ *bits* to her observers.



Natural Algorithms for Plurality Consensus



Natural Algorithms for Plurality Consensus



Dynamics

(informal) *Very simple* distributed algorithms: For every graph $G = (V, E)$, agent $u \in V$ and round $t \in \mathbb{N}$, states are updated according to fixed rule $f(\sigma(u), \sigma(S))$ of current state $\sigma(u)$ and symmetric function of states $\sigma(S)$ of a random sample S of neighbors.

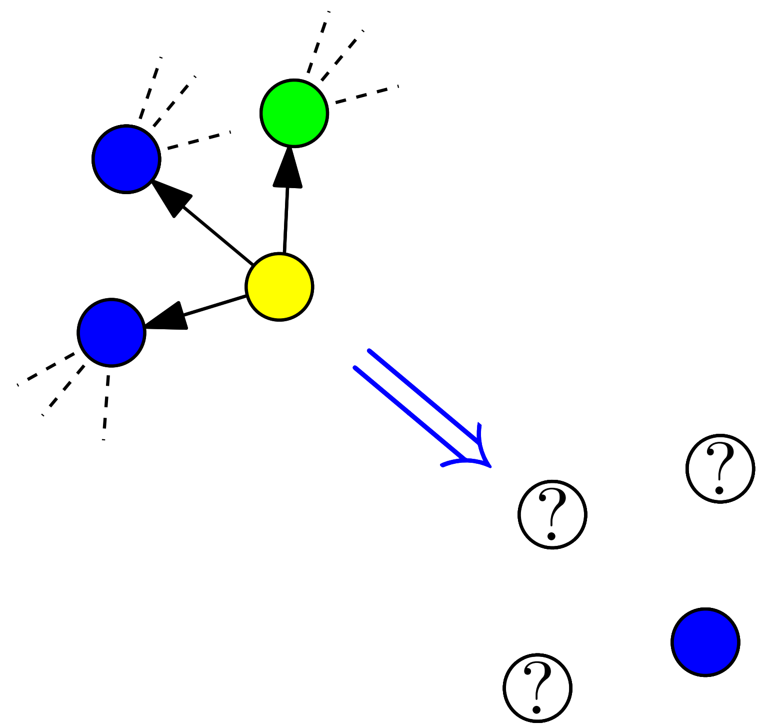
Dynamics

(informal)

Very simple distributed algorithms: For every graph $G = (V, E)$, agent $u \in V$ and round $t \in \mathbb{N}$, states are updated according to fixed rule $f(\sigma(u), \sigma(S))$ of current state $\sigma(u)$ and symmetric function of states $\sigma(S)$ of a random sample S of neighbors.

Two examples:

- 3-Majority dynamics



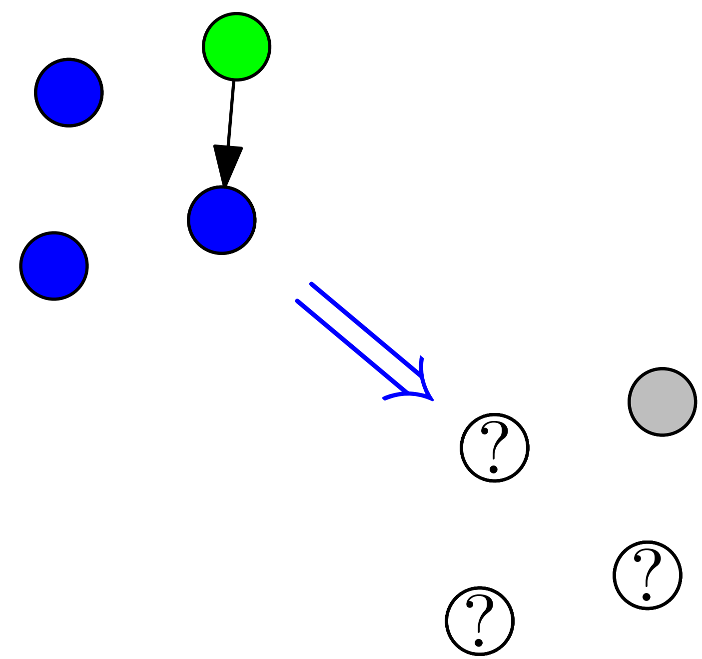
Dynamics

(informal)

Very simple distributed algorithms: For every graph $G = (V, E)$, agent $u \in V$ and round $t \in \mathbb{N}$, states are updated according to fixed rule $f(\sigma(u), \sigma(S))$ of current state $\sigma(u)$ and symmetric function of states $\sigma(S)$ of a random sample S of neighbors.

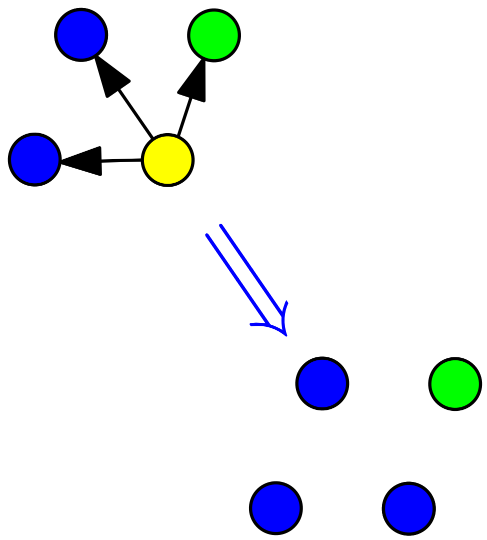
Two examples:

- 3-Majority dynamics
- Undecided-state dynamics



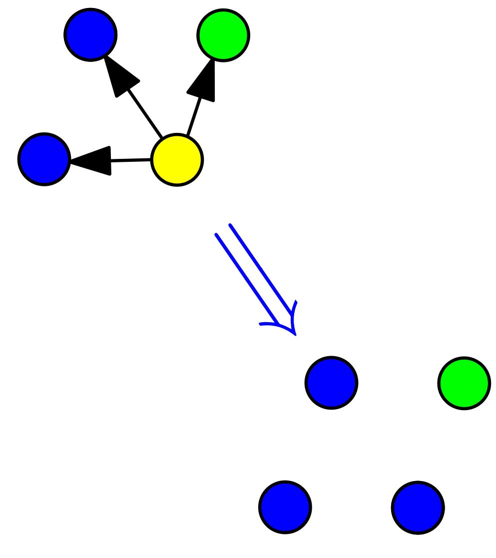
The Power of Dynamics: Plurality Consensus

3-Majority dynamics [SPAA '14, SODA '16]. If plurality has **bias** $\mathcal{O}(\sqrt{kn \log n})$, converges to it in $\mathcal{O}(k \log n)$ rounds w.h.p., even against $o(\sqrt{n/k})$-bounded adversary. Without bias, converges in poly(k). h-majority converges in $\Omega(k/h^2)$.

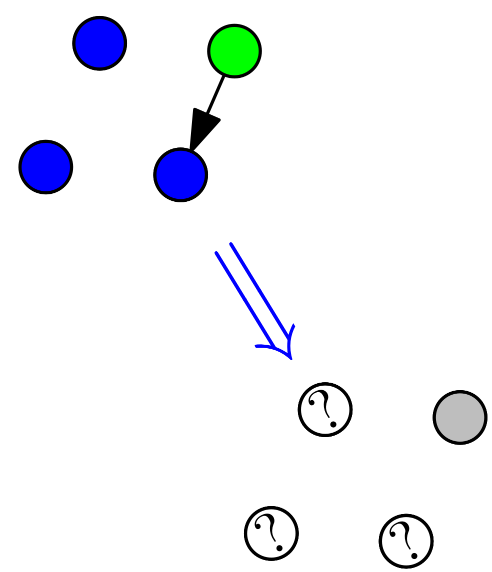


The Power of Dynamics: Plurality Consensus

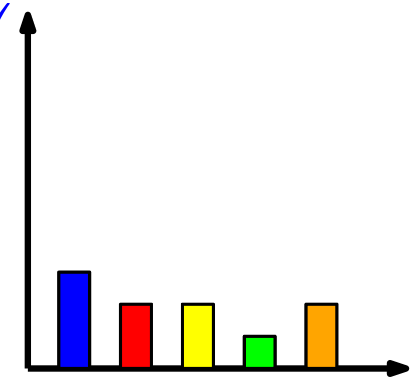
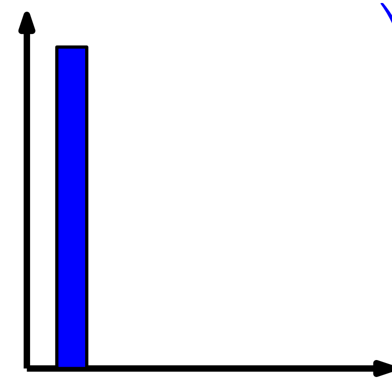
3-Majority dynamics [SPAA '14, SODA '16]. If plurality has **bias** $\mathcal{O}(\sqrt{kn\log n})$, converges to it in $\mathcal{O}(k\log n)$ rounds w.h.p., even against $o(\sqrt{n/k})$-bounded adversary. Without bias, converges in poly(k). h-majority converges in $\Omega(k/h^2)$.

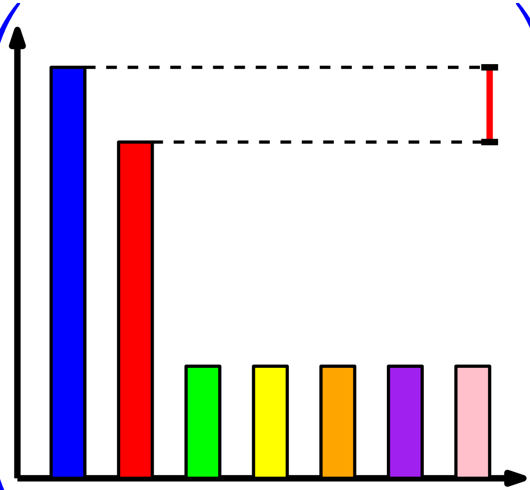
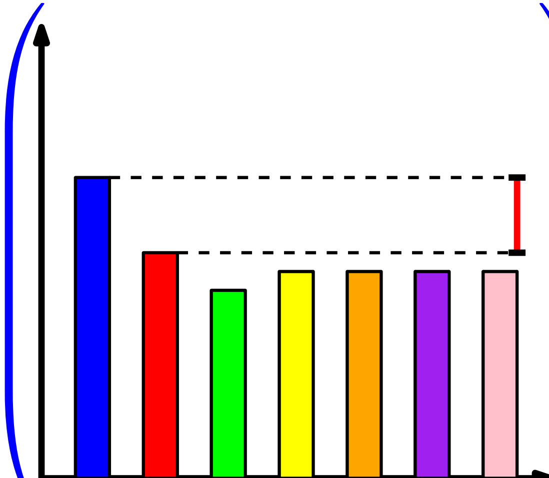


Undecided-State dynamics [SODA '15]. If majority/second-majority $(c_{maj}/c_{2^{nd}maj})$ is at least $1+\epsilon$, system converges to plurality within $\tilde{\Theta}(\mathrm{md}(\mathbf{c}))$ rounds w.h.p.



A Global Measure of Bias

$$\mathrm{md}(\mathbf{c}^{(\mathbf{0})}) := \sum_{i=1}^{k} \left(\frac{c_i^{(0)}}{c_{maj}^{(0)}} \right)^2 = 1 + \mathcal{D} \Bigg($$ ,  $\Bigg)$

$1 \leq \mathrm{md} \Bigg($  $\Bigg) \ll \mathrm{md} \Bigg($  $\Bigg) \leq k$

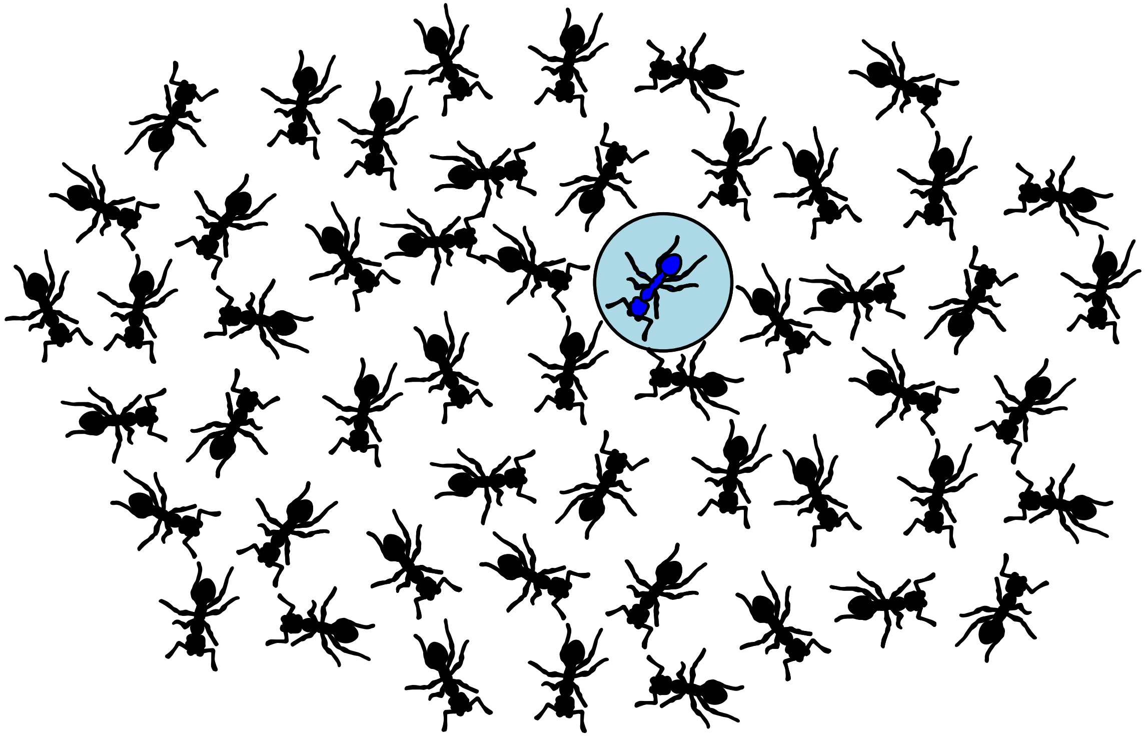
Undecided-State dynamics [SODA '15]. If majority/second-majority ($c_{maj}/c_{2^{nd}maj}$) is at least $1+\epsilon$, system converges to plurality within $\tilde{\Theta}(\mathrm{md}(\mathbf{c}))$ rounds w.h.p.

Applications: Broadcast Problem

Applications: Broadcast Problem

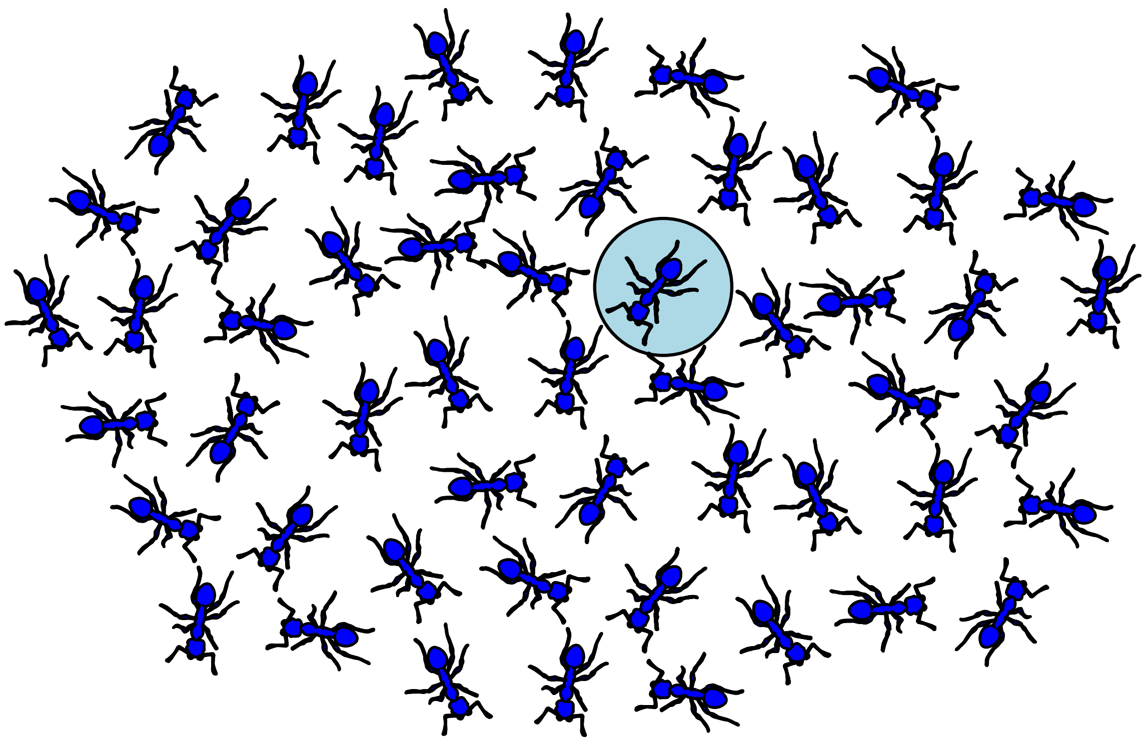
Applications: (Self-stabilizing) Broadcast Problem

Sources' bits (and other agents' states) may change in response to *external environment*.



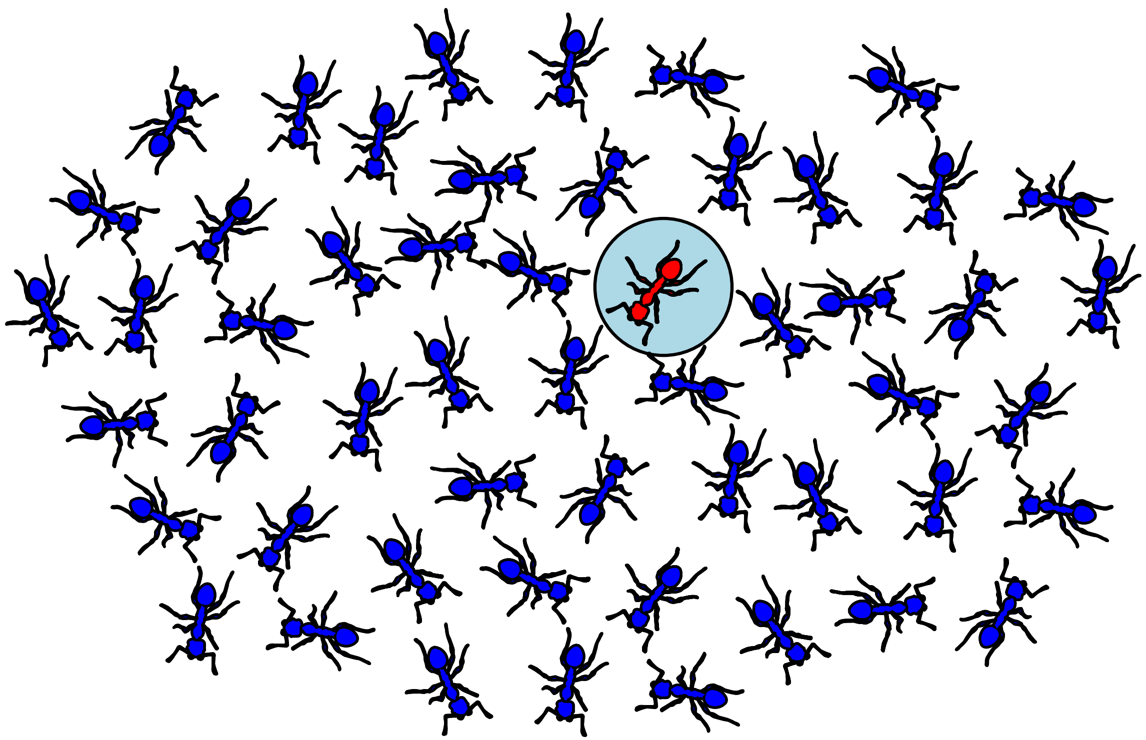
Applications: (Self-stabilizing) Broadcast Problem

Sources' bits (and other agents' states) may change in response to *external environment*.



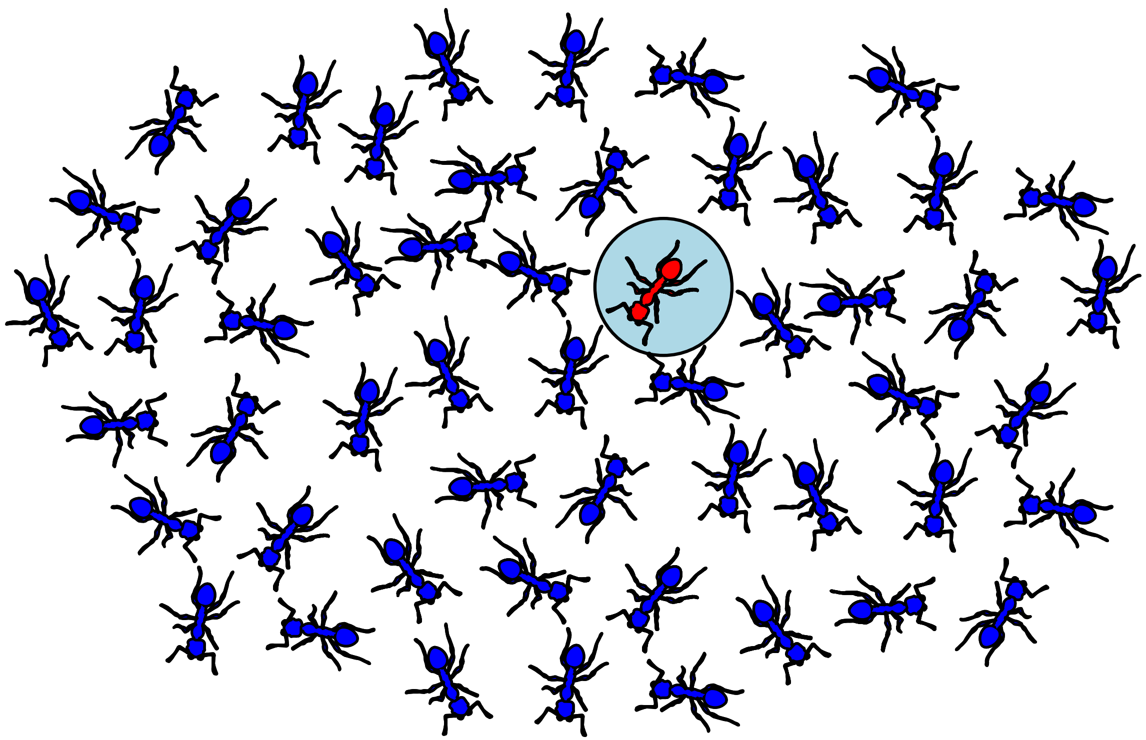
Applications: (Self-stabilizing) Broadcast Problem

Sources' bits (and other agents' states) may change in response to *external environment*.



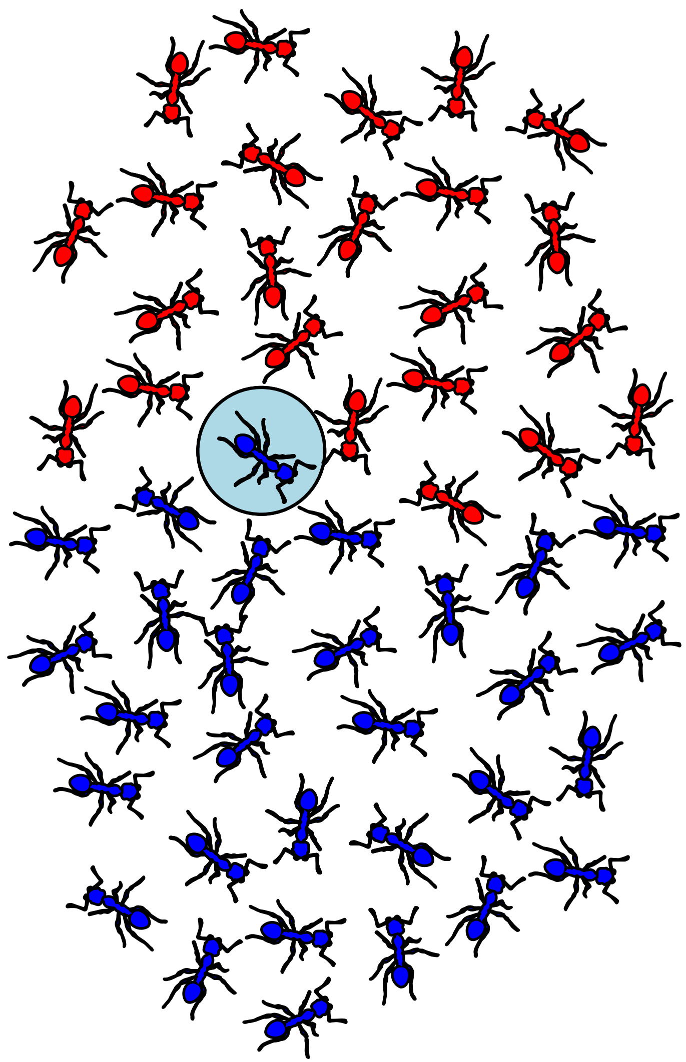
Applications: (Self-stabilizing) Broadcast Problem

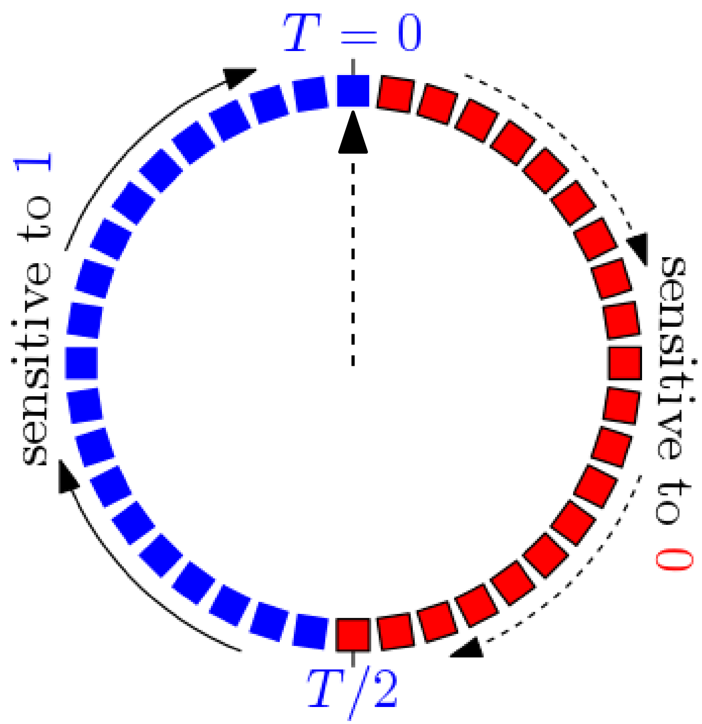
Sources' bits (and other agents' states) may change in response to *external environment*.



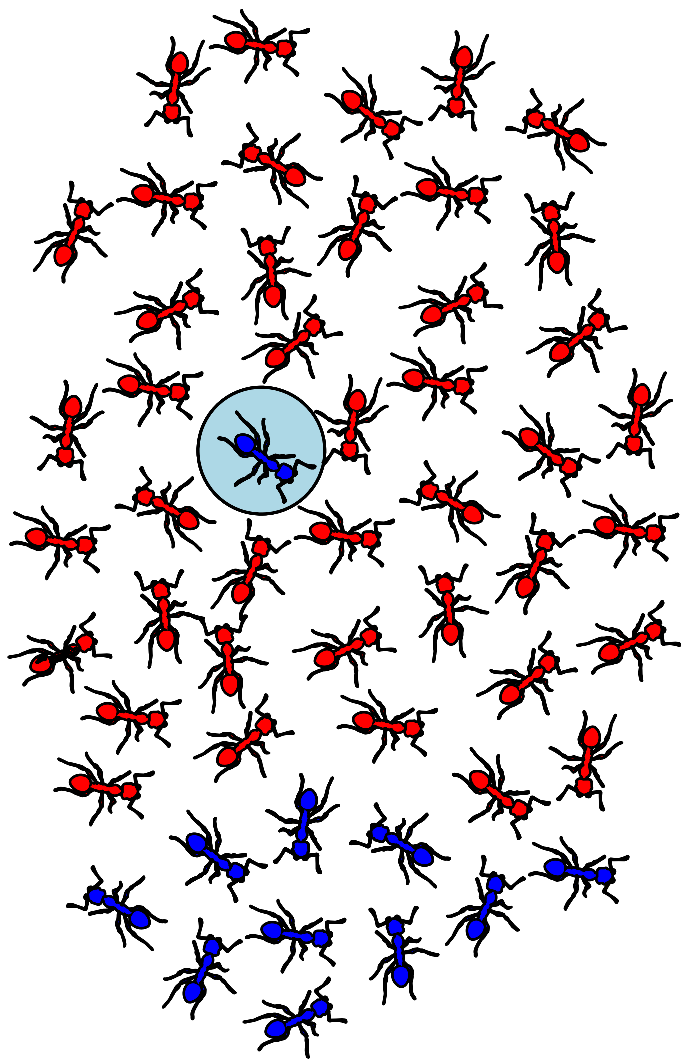
More generally, system is initialized in *arbitrary state* (self-stabilization).

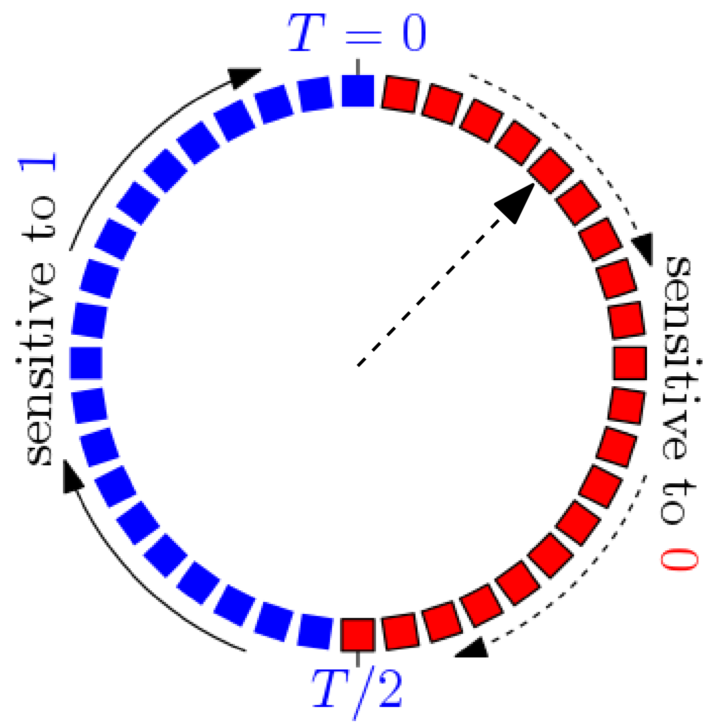
(Self-Stab.) Broadcast vs Synchronization



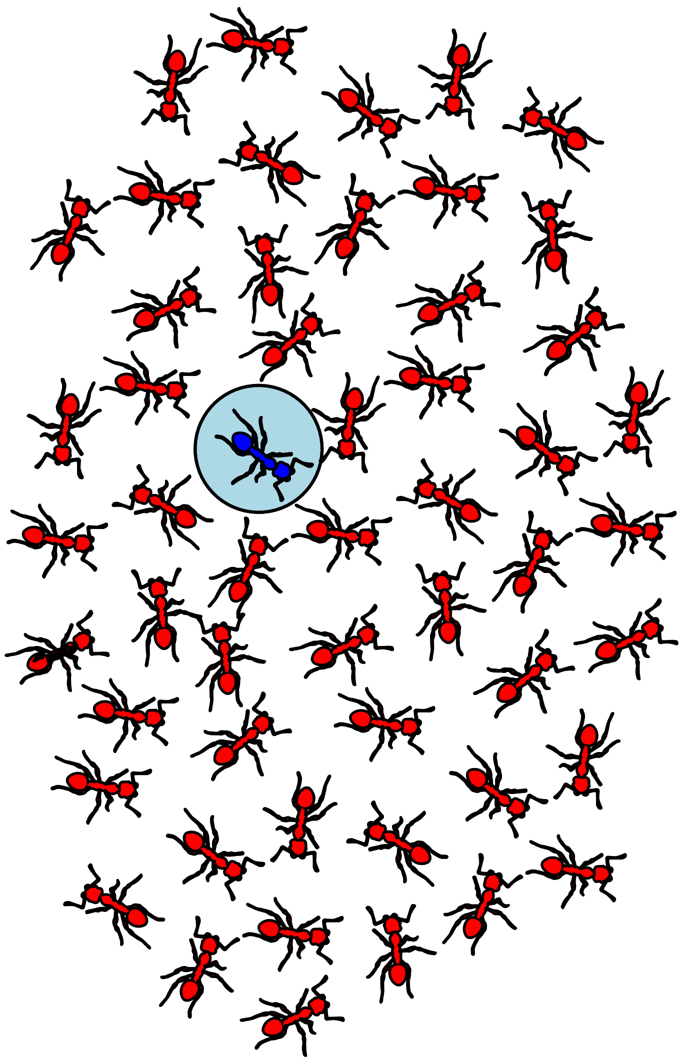


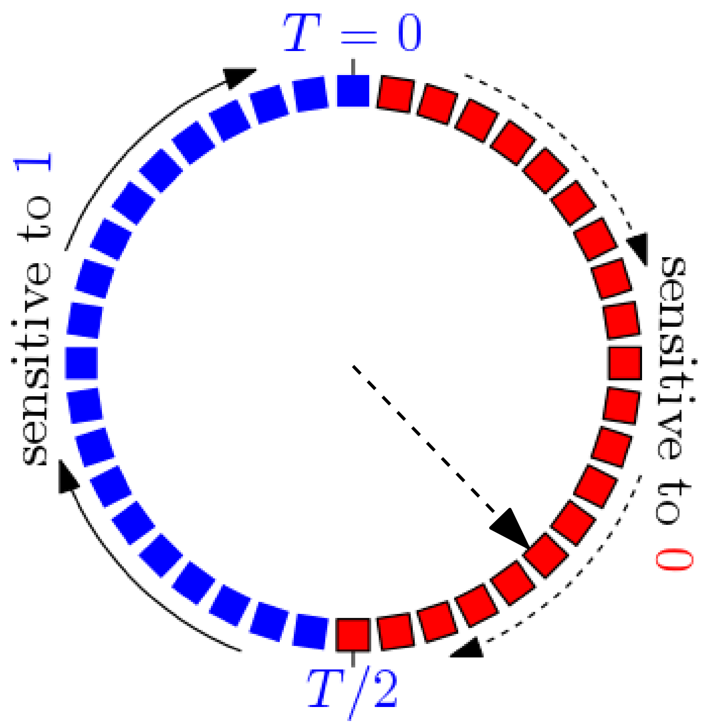
(Self-Stab.) Broadcast vs Synchronization



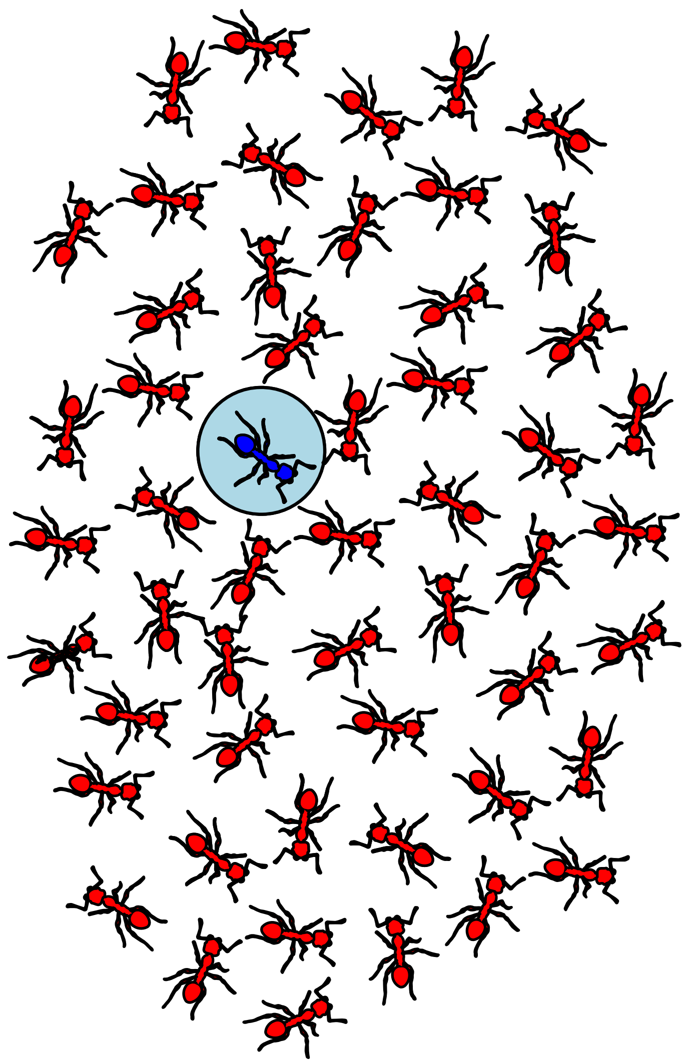


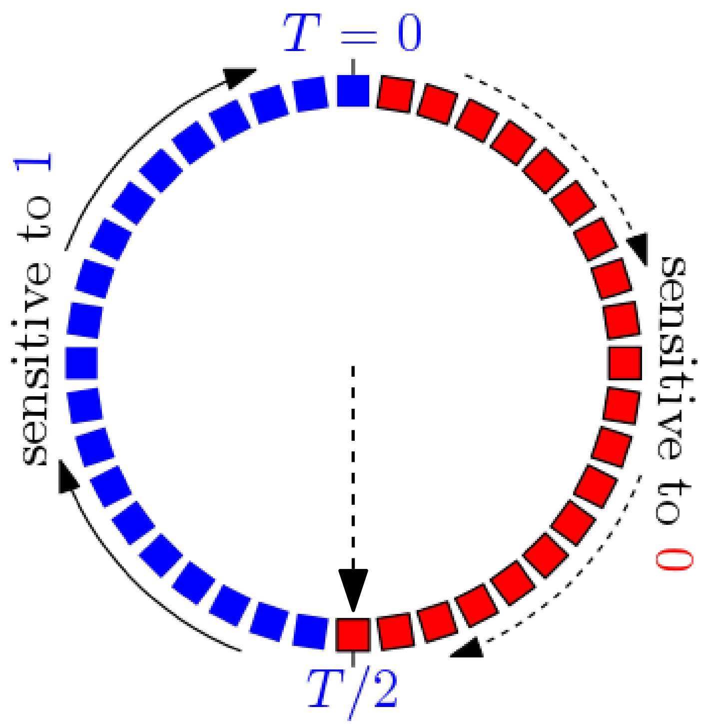
(Self-Stab.) Broadcast vs Synchronization



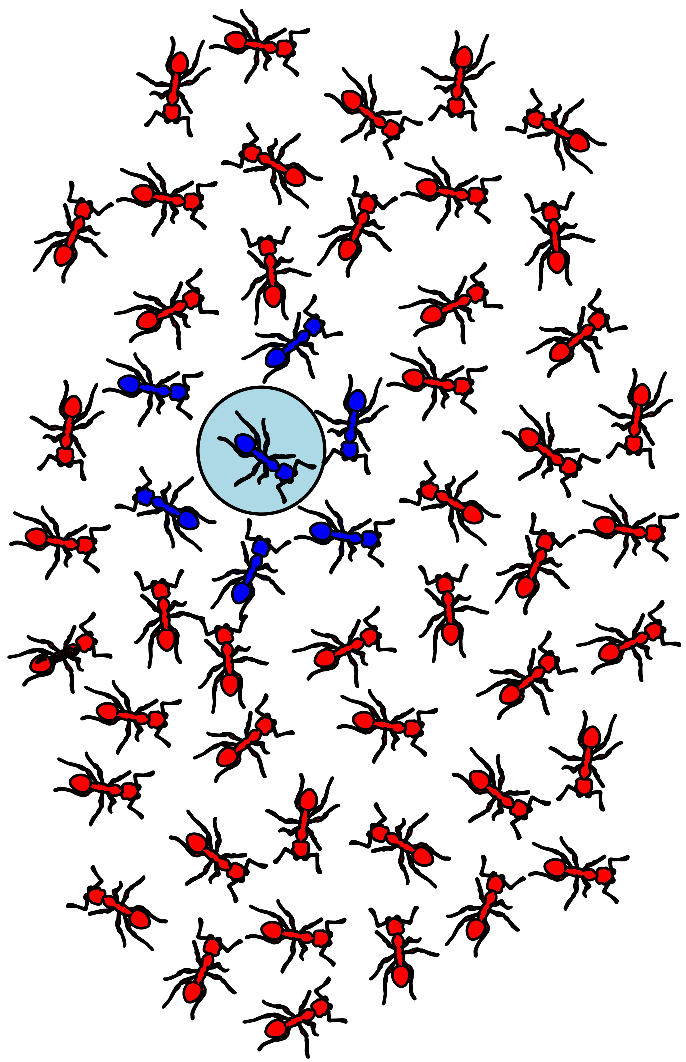


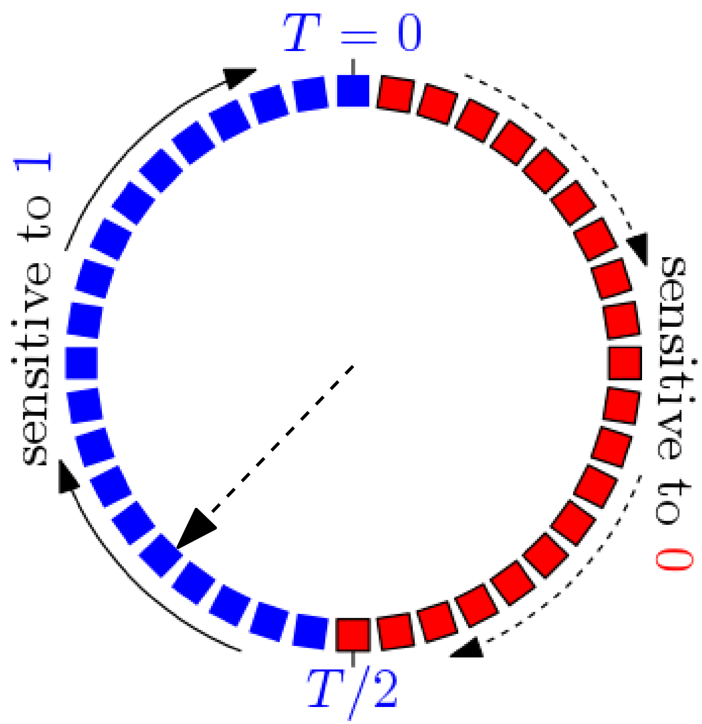
(Self-Stab.) Broadcast vs Synchronization



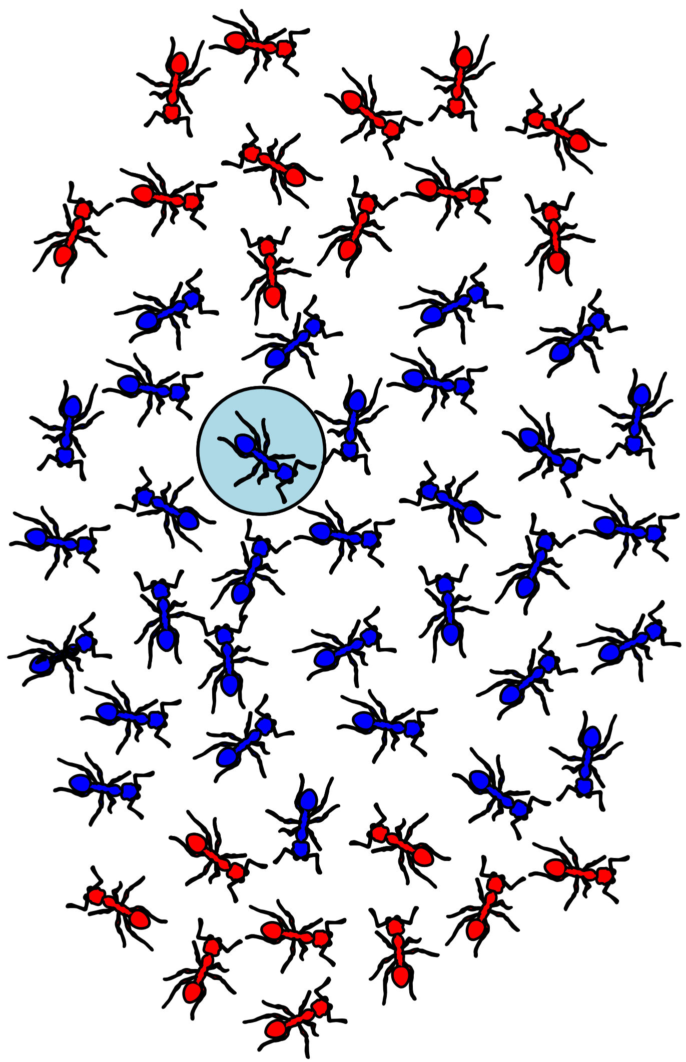


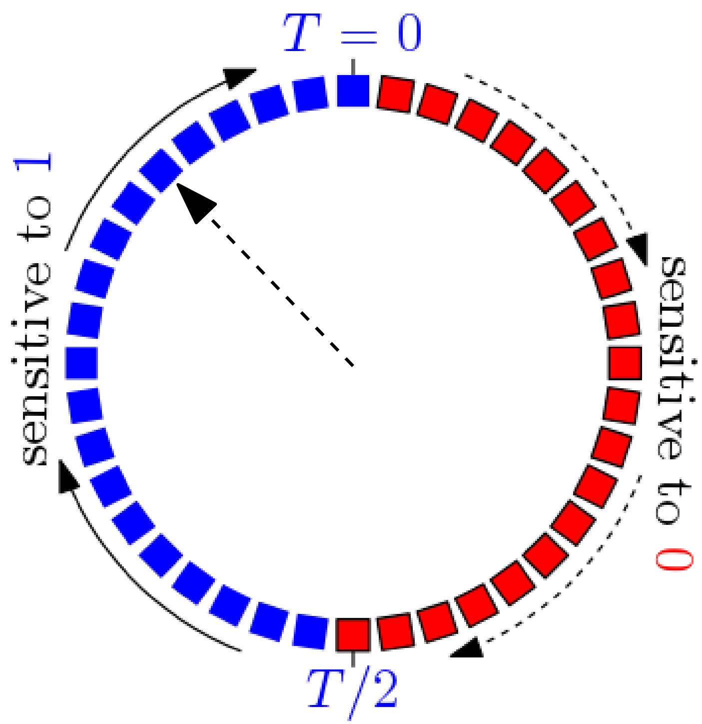
(Self-Stab.) Broadcast vs Synchronization



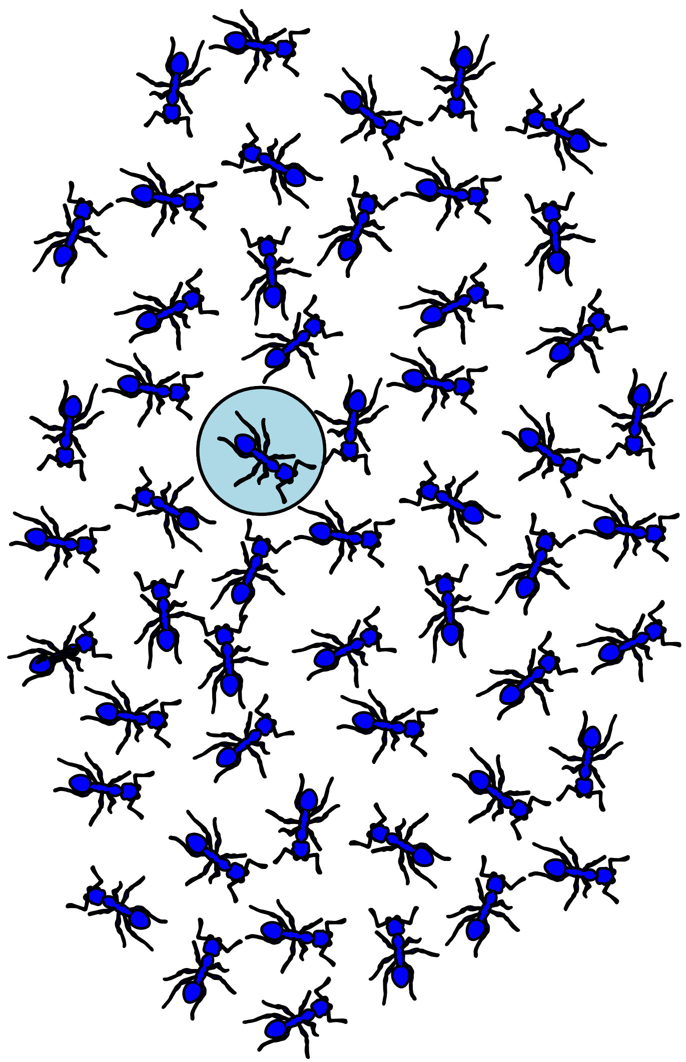


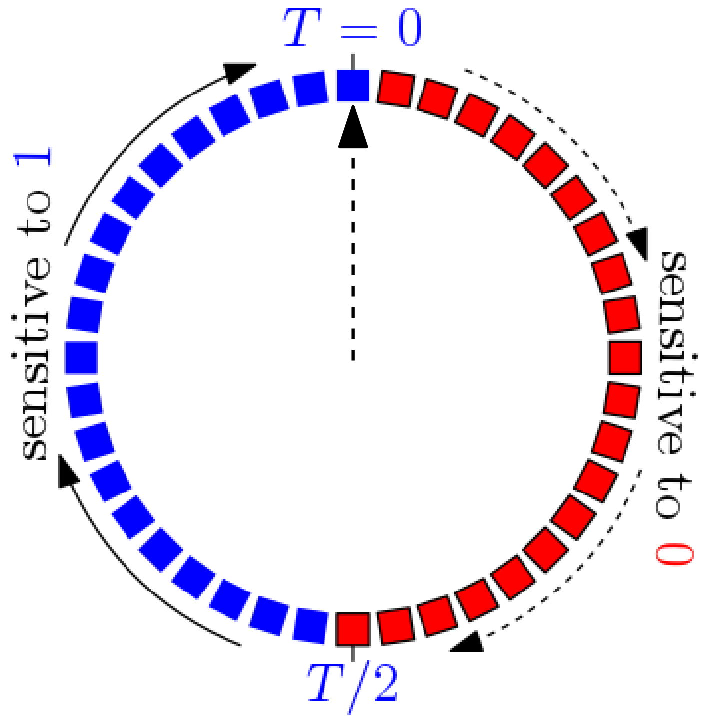
(Self-Stab.) Broadcast vs Synchronization





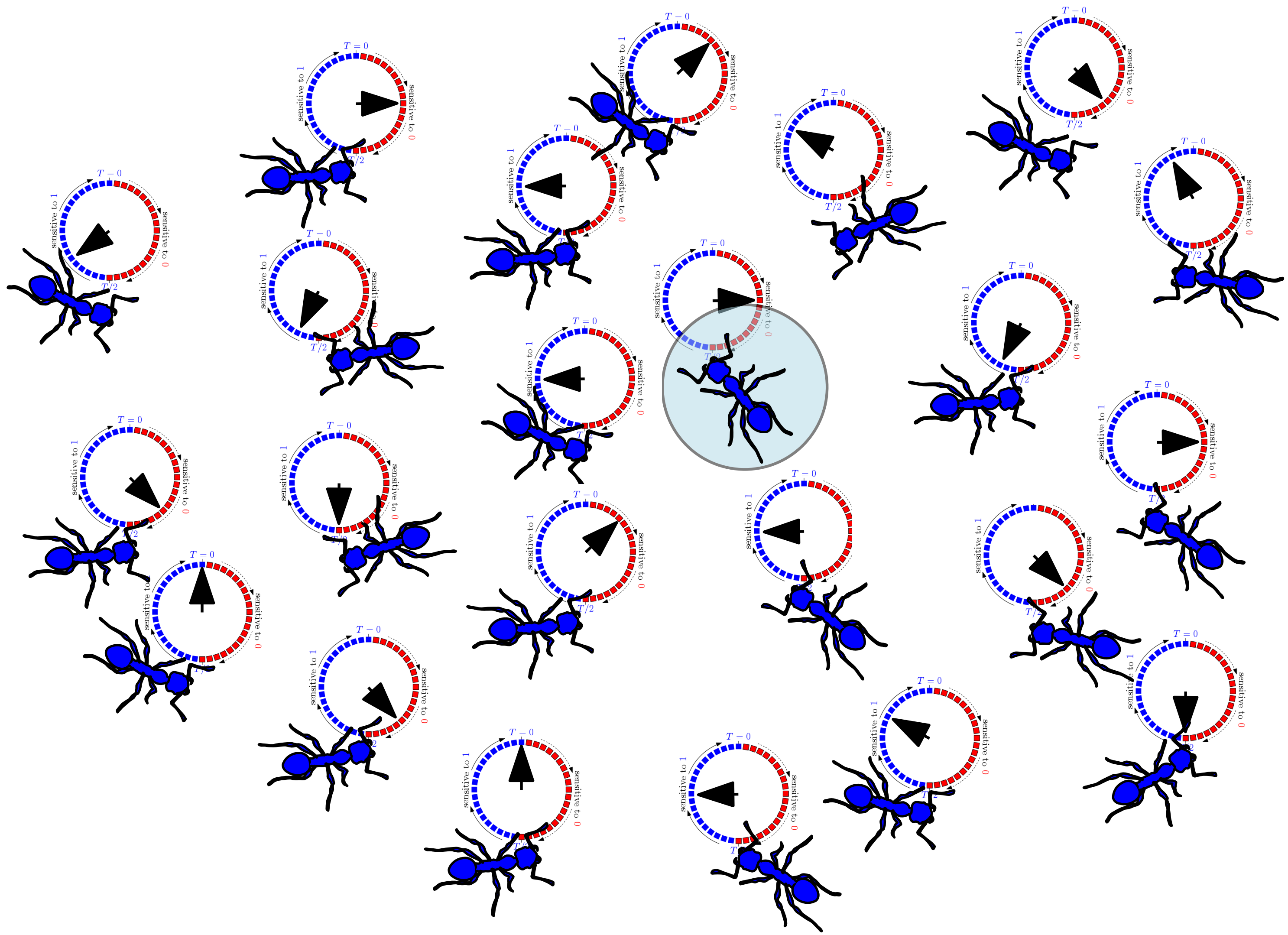
(Self-Stab.) Broadcast vs Synchronization



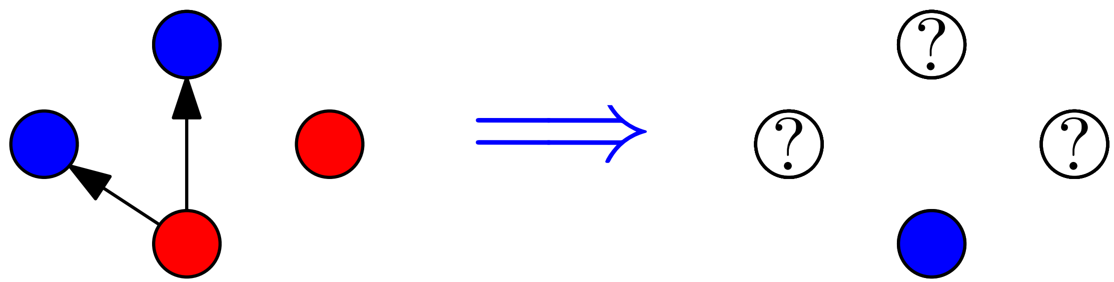


(Self-Stab.) Broadcast vs Synchronization

Self-stablizing algorithms converge from
any initial configuration

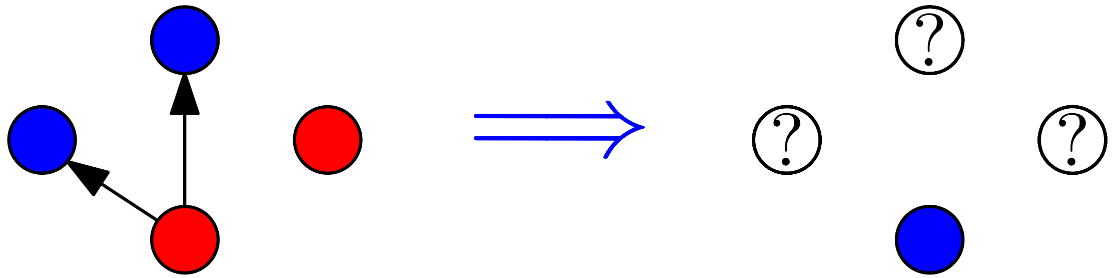


Self-Stabilizing Clock Sync. in the $\mathcal{PULL}$ Model

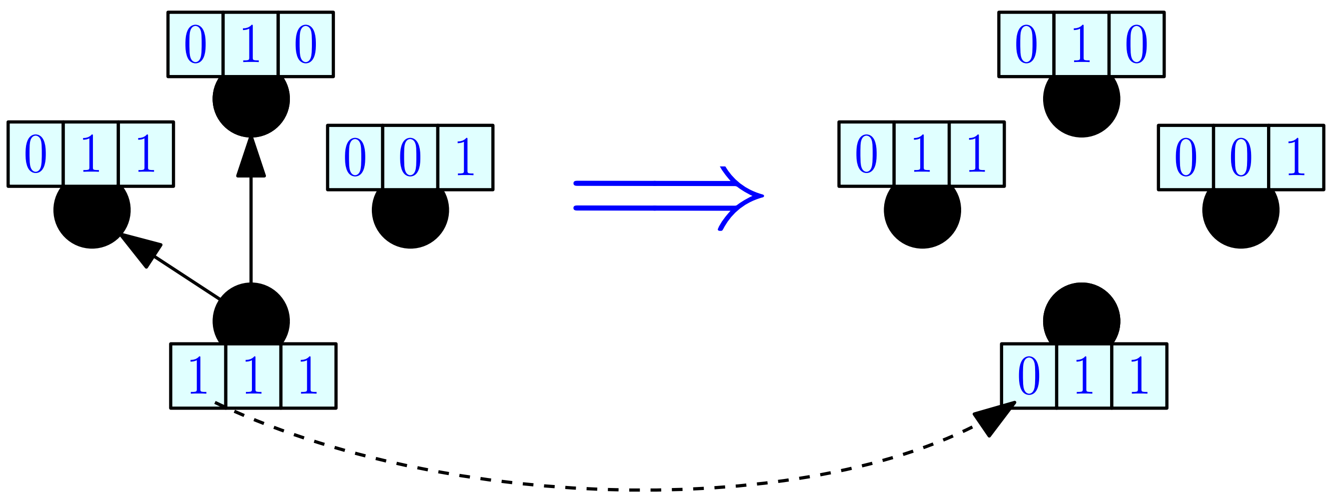


2-Majority dynamics [Doerr et al. '11]. Converge to consensus in $\mathcal{O}(\log n)$ rounds with high probability.

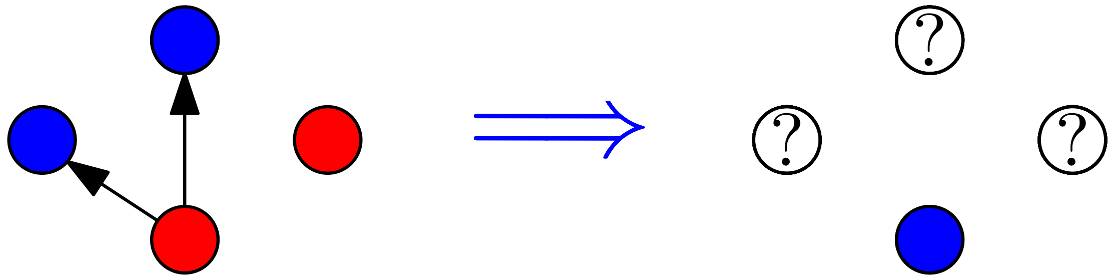
Self-Stabilizing Clock Sync. in the $\mathcal{PULL}$ Model



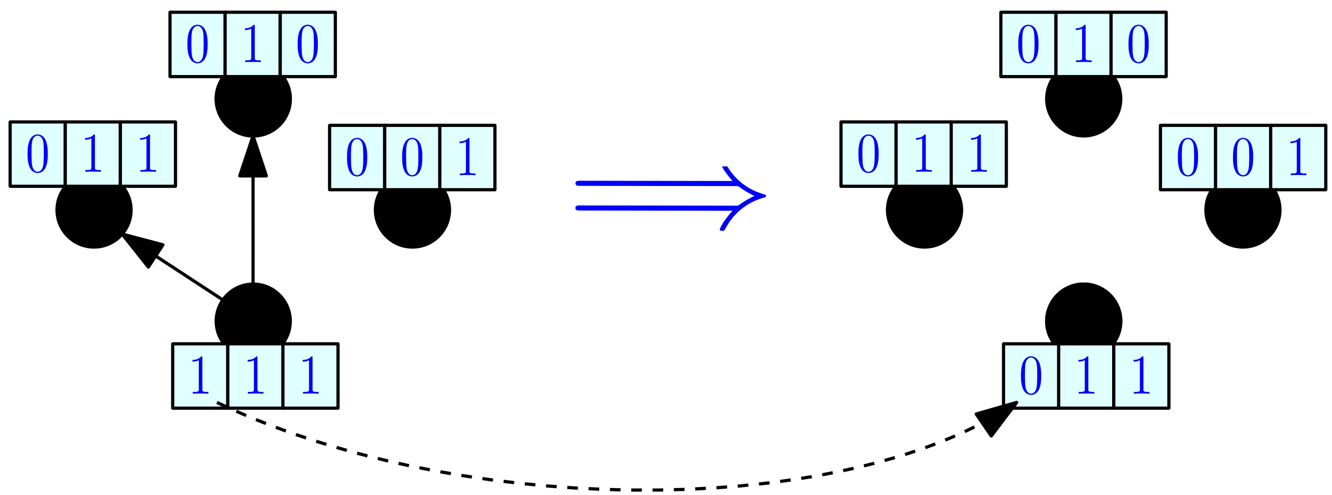
2-Majority dynamics [Doerr et al. '11]. Converge to consensus in $\mathcal{O}(\log n)$ rounds with high probability.



Self-Stabilizing Clock Sync. in the $\mathcal{PULL}$ Model

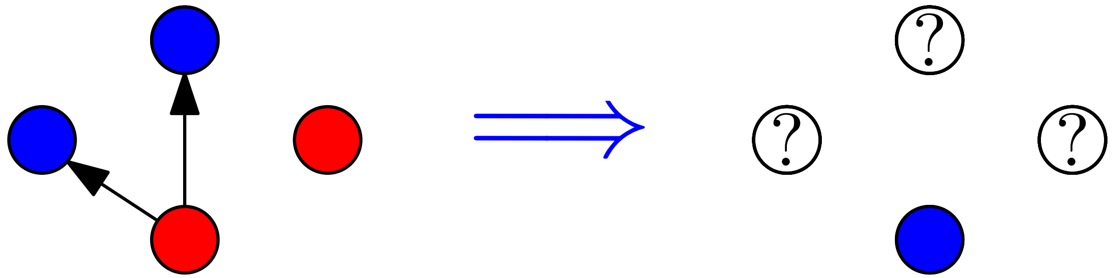


2-Majority dynamics [Doerr et al. '11]. Converge to consensus in $\mathcal{O}(\log n)$ rounds with high probability.

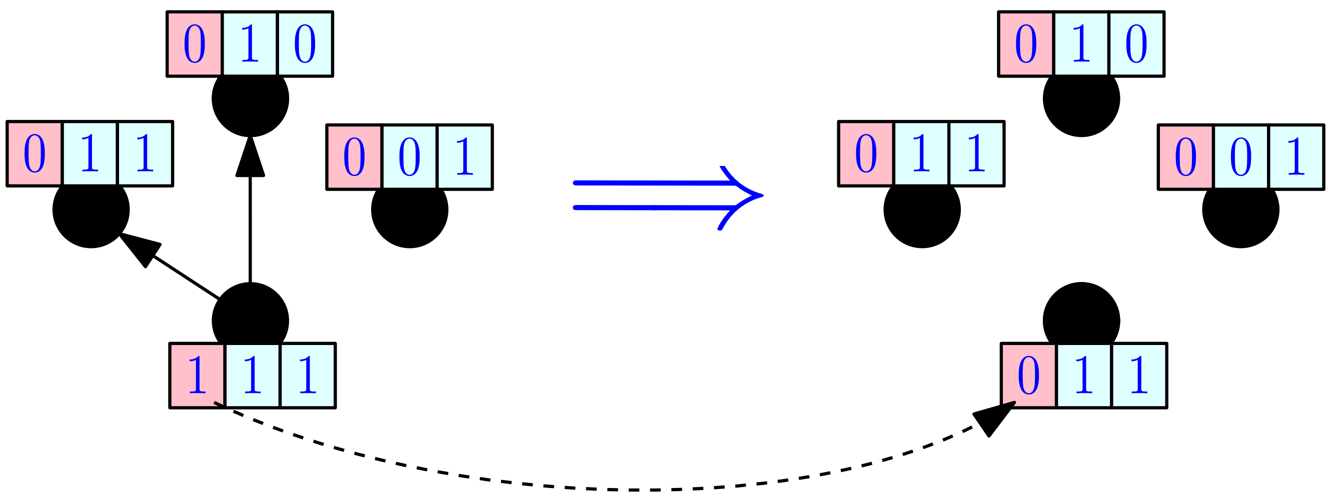


T-clock can be sync. in $\mathcal{O}(\log n \log T)$ rounds w.h.p. using $\log T$ bits. Binary broadcast can be done in 1-*bit* $\mathcal{PULL}$...

Self-Stabilizing Clock Sync. in the $\mathcal{PULL}$ Model

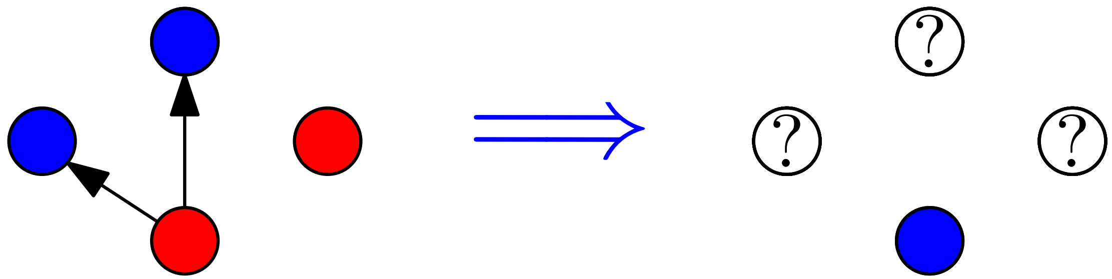


2-Majority dynamics [Doerr et al. '11]. Converge to consensus in $\mathcal{O}(\log n)$ rounds with high probability.

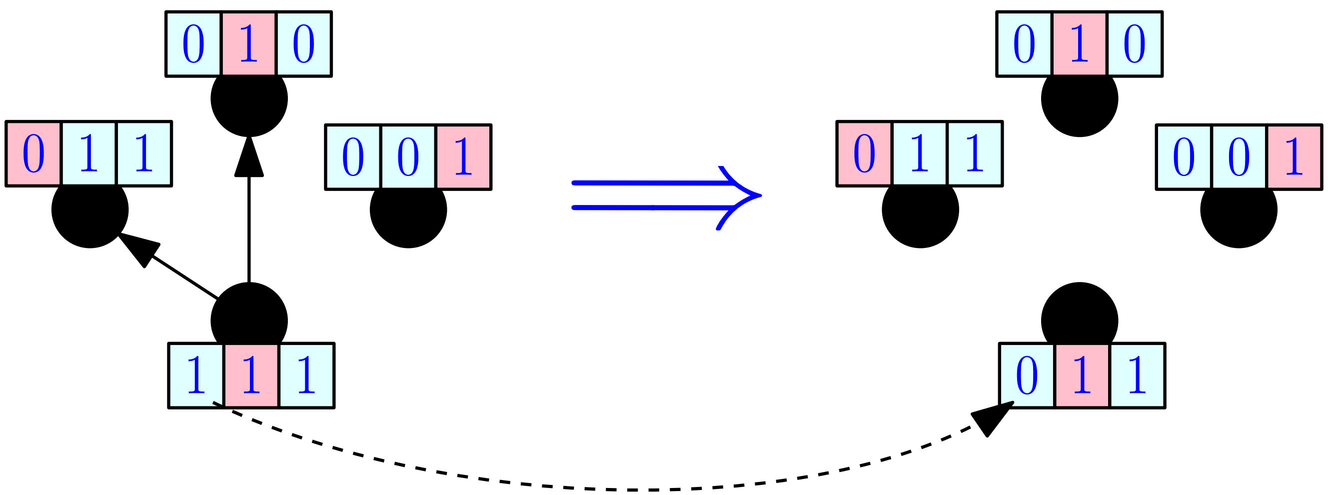


T-clock can be sync. in $\mathcal{O}(\log n \log T)$ rounds w.h.p. using $\log T$ bits. Binary broadcast can be done in 1-*bit* $\mathcal{PULL}$...

Self-Stabilizing Clock Sync. in the $\mathcal{PULL}$ Model

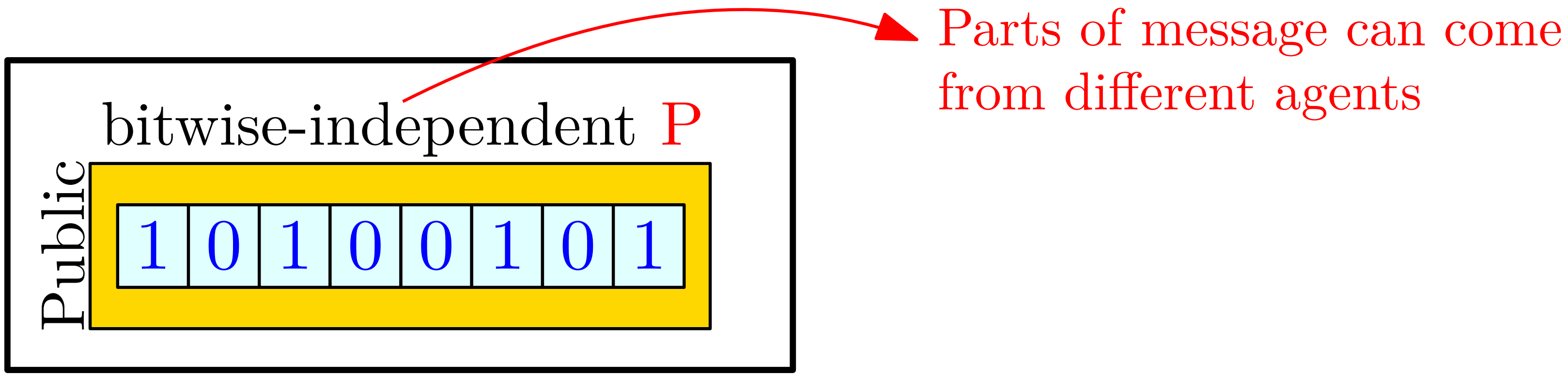


2-Majority dynamics [Doerr et al. '11]. Converge to consensus in $\mathcal{O}(\log n)$ rounds with high probability.

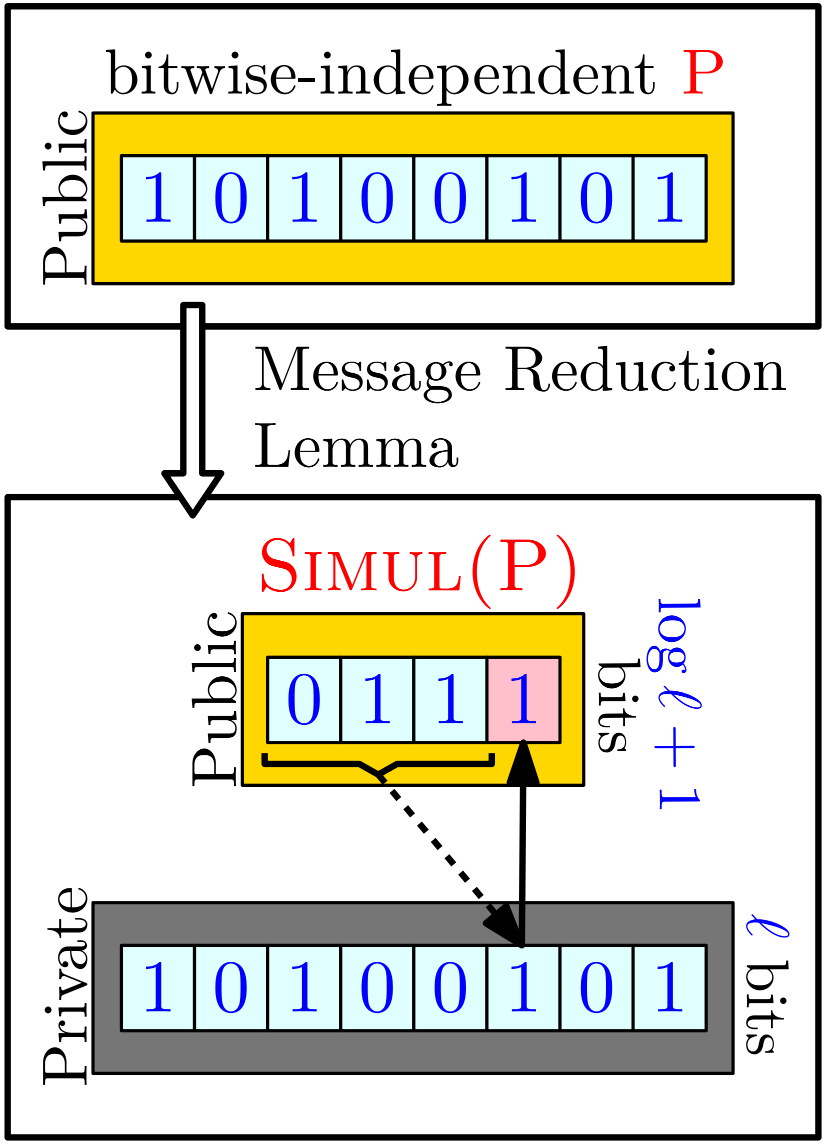


T-clock can be sync. in $\mathcal{O}(\log n \log T)$ rounds w.h.p. using $\log T$ bits. Binary broadcast can be done in 1-*bit* $\mathcal{PULL}$...

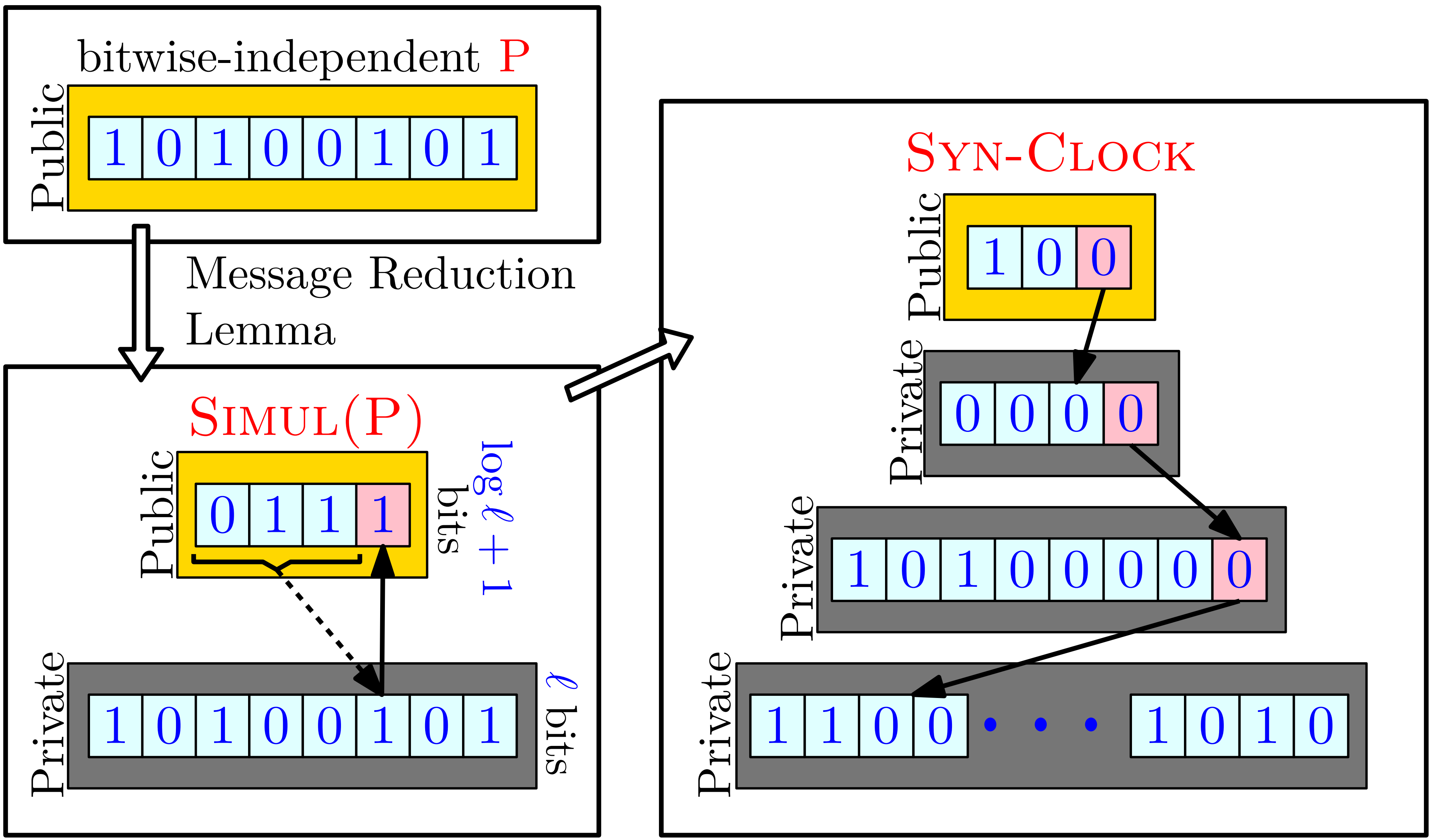
The Message Reduction Lemma



The Message Reduction Lemma



The Message Reduction Lemma



Results: 3 Bits suffice...

Theorem (Clock Syncronization). SYN-CLOCK is a *self-stabilizing* clock synchronization protocol which synchronizes a clock modulo T in $\tilde{\mathcal{O}}(\log n \log T)$ rounds w.h.p. using **3-bit messages**.

Corollary (Self-stabilizing Majority Broadcast). SYN-PHASE-SPREAD is a *self-stabilizing* Majority Broadcast protocol which converges in $\tilde{\mathcal{O}}(\log n)$ rounds w.h.p using **3-bit messages**, provided majority is supported by $(\frac{1}{2} + \epsilon)$-fraction of source agents.

Conclusions

Biology demands the study of systems in-between interacting-particle systems and human-made ones.

TCS can analyze natural algorithms, helping to understand principles behind the systems' ability to compute in simple chaotic ways.

Thank You!