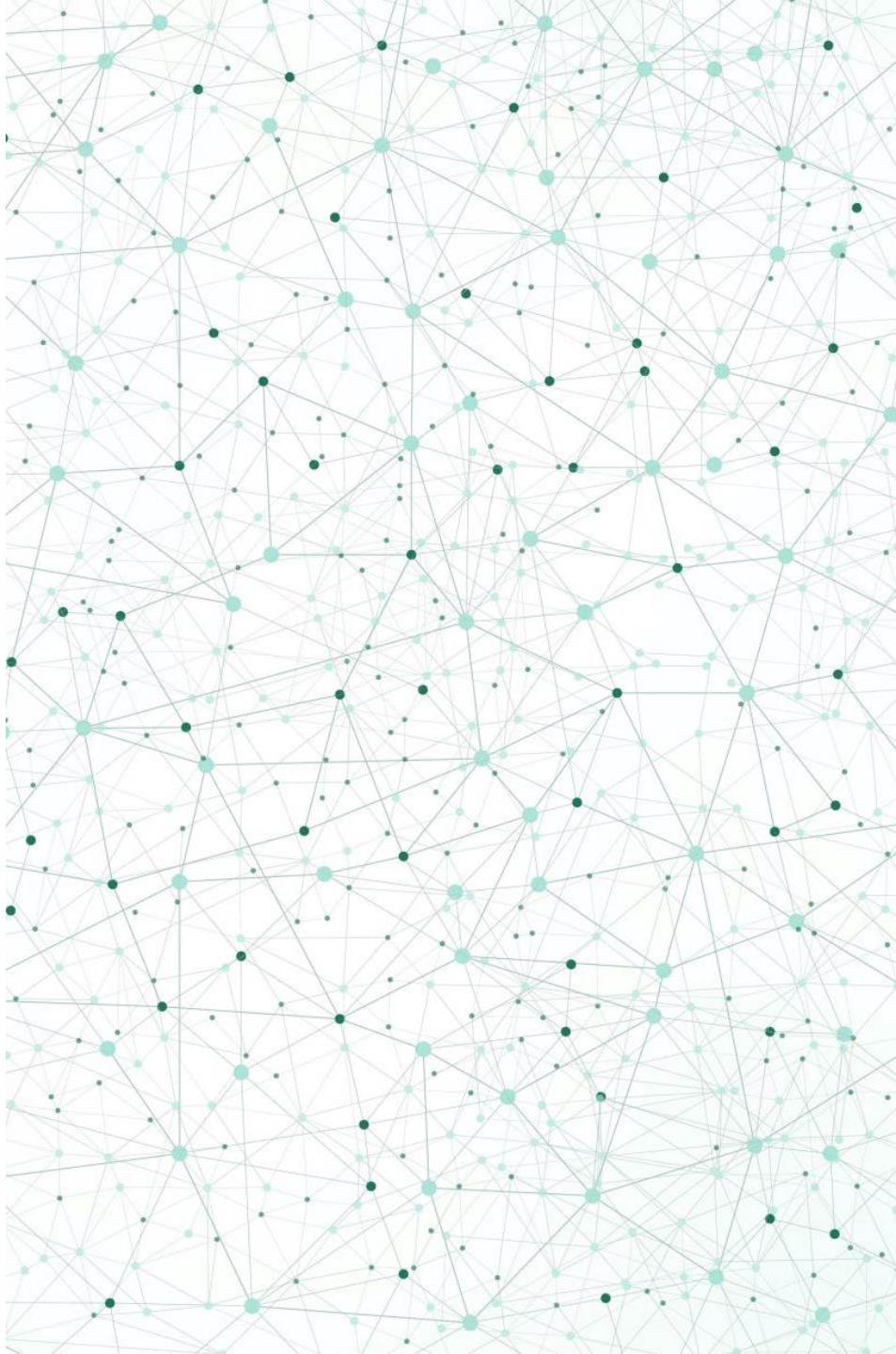
Adaptive Parallelization of Multi-Agent Simulations with Localized Dynamics

Alexandru-Ionuț Băbeanu, Tatiana Filatova,
Jan H. Kwakkel, Neil Yorke-Smith

The 24th International Workshop on
Multi-Agent-Based Simulation

30 May 2023

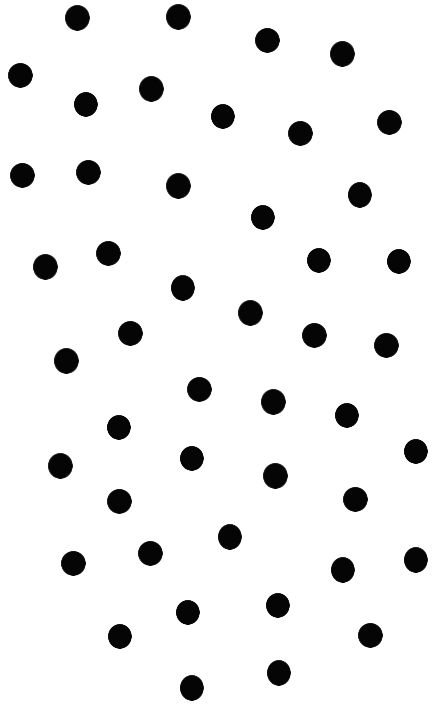


TU Delft

TAILOR

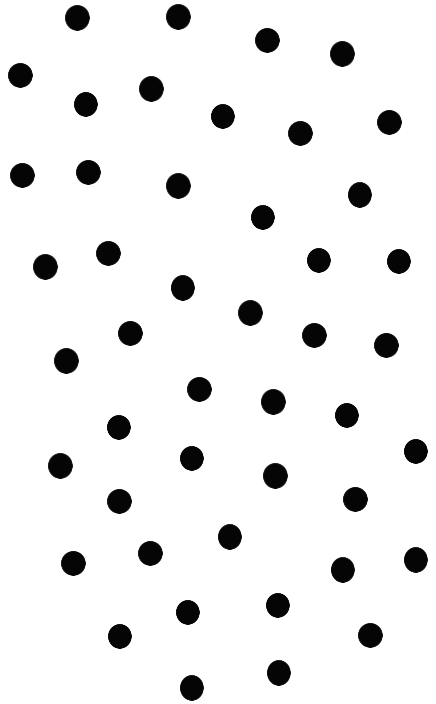
Motivation

- MABS are computationally very demanding



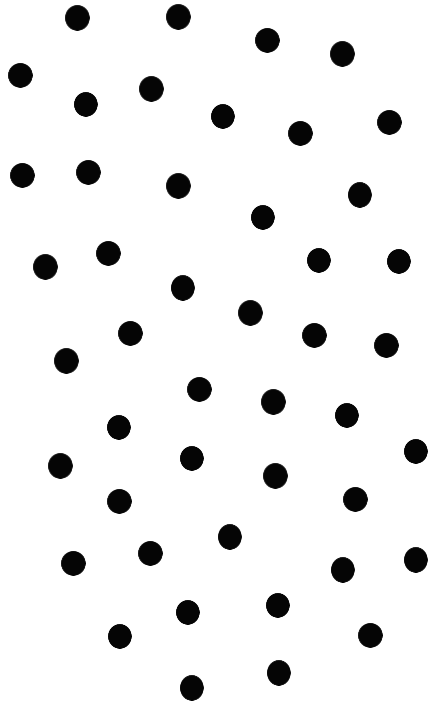
Motivation

- MABS are computationally very demanding
- Realistic simulations may require billions of agents



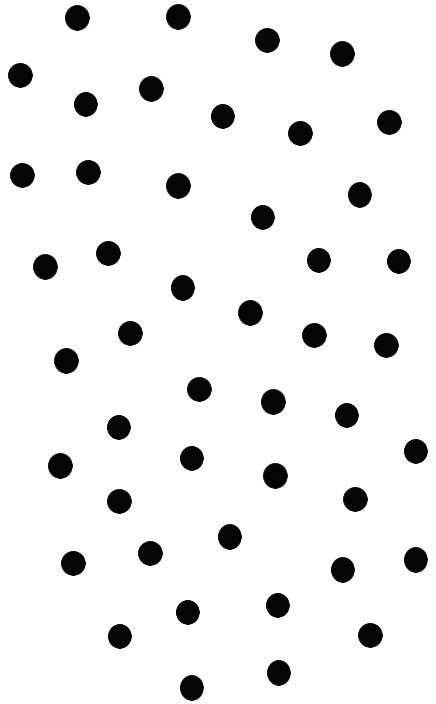
Motivation

- MABS are computationally very demanding
- Realistic simulations may require billions of agents
- One acceleration strategy combines:
  - System modularity
  - Multicore parallelism



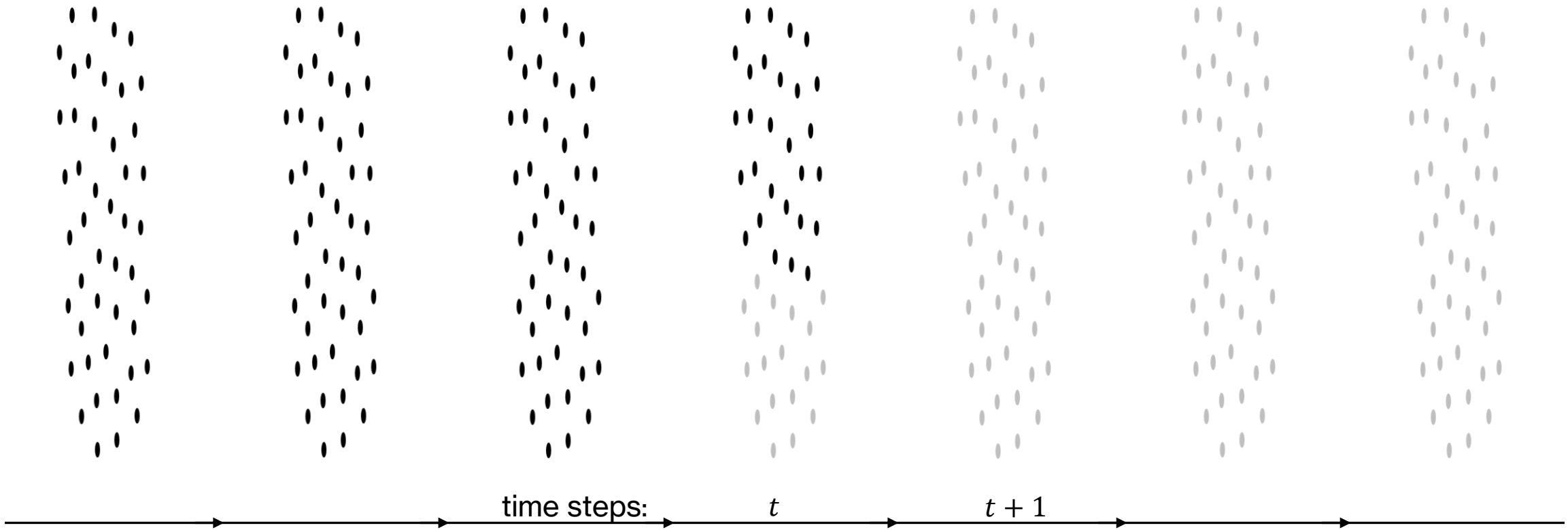
Motivation

- MABS are computationally very demanding
- Realistic simulations may require billions of agents
- One acceleration strategy combines:
  - System modularity
  - Multicore parallelism
- But there are complications:
  - Shared information, interactions
  - synchronization-related waiting



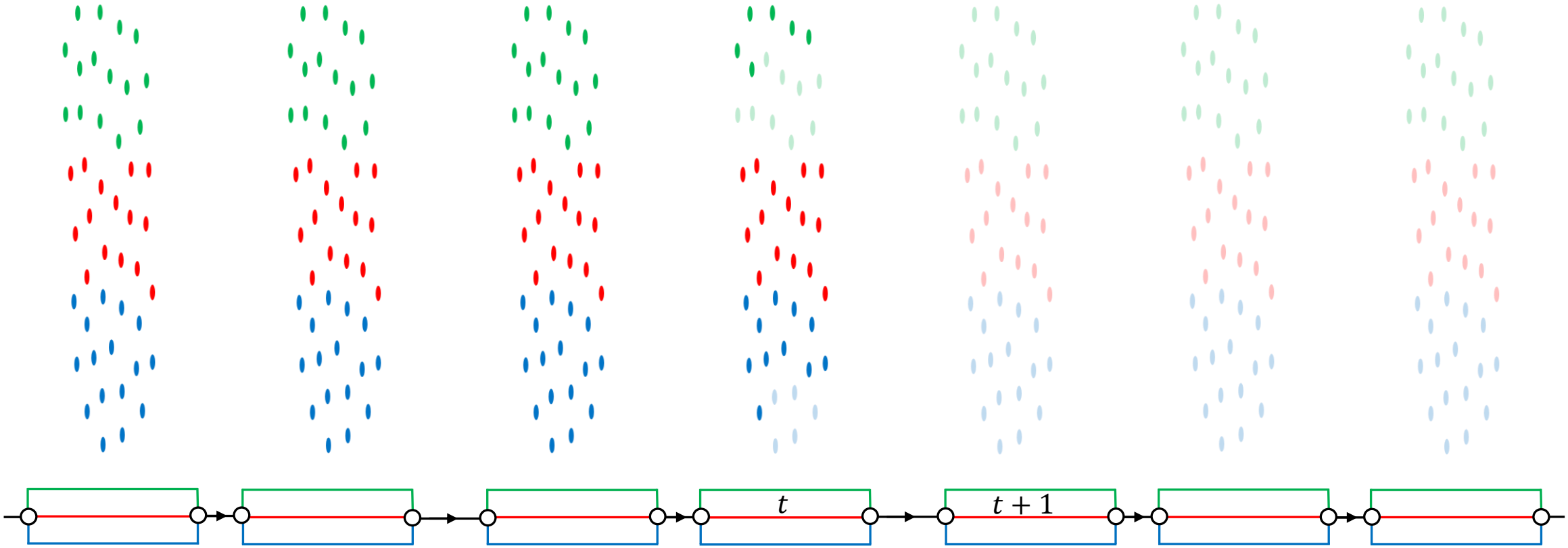
General intuition

- No parallelization:



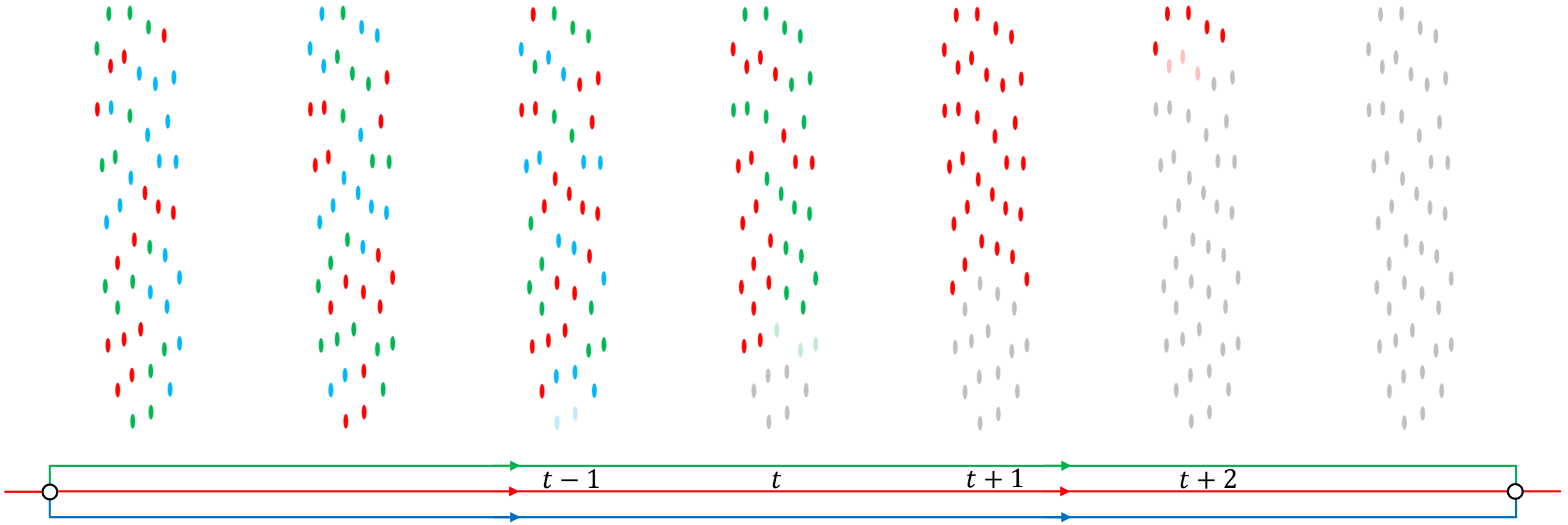
General intuition

- Conventional parallelization (rigid, top-down):



General intuition

- Desired parallelization (adaptive, bottom-up):

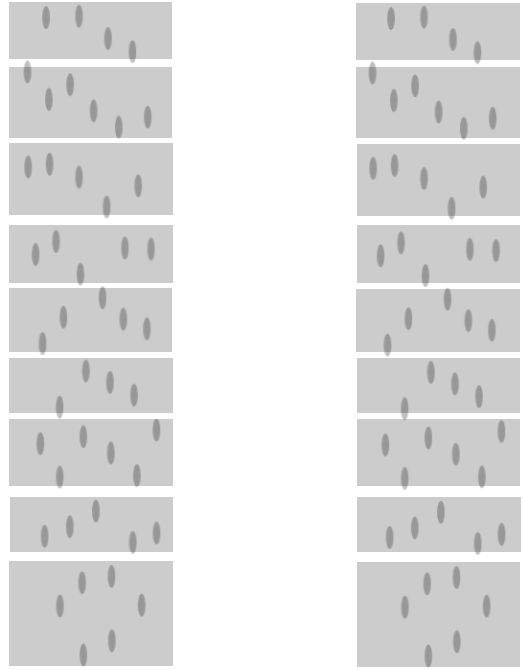


An adaptive parallelization protocol

- Need-based preregistration of updates

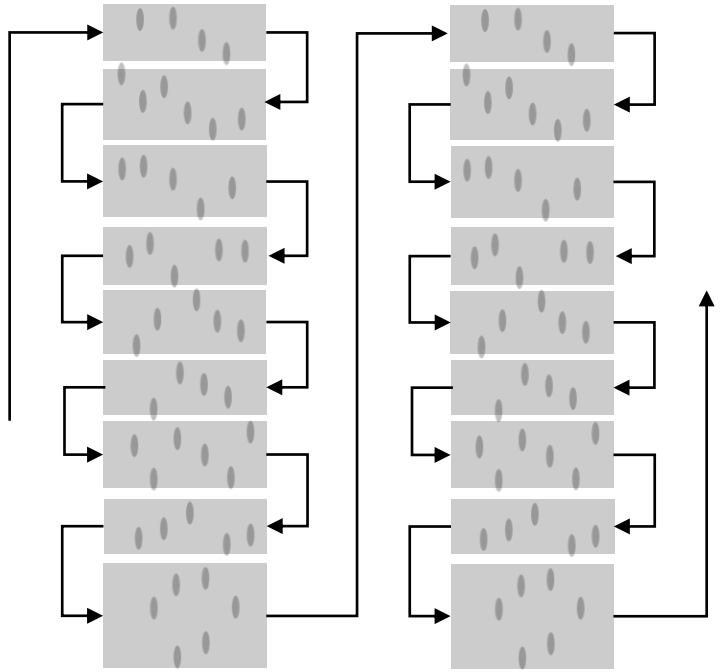
An adaptive parallelization protocol

- Need-based preregistration of updates
- Updates grouped into tasks



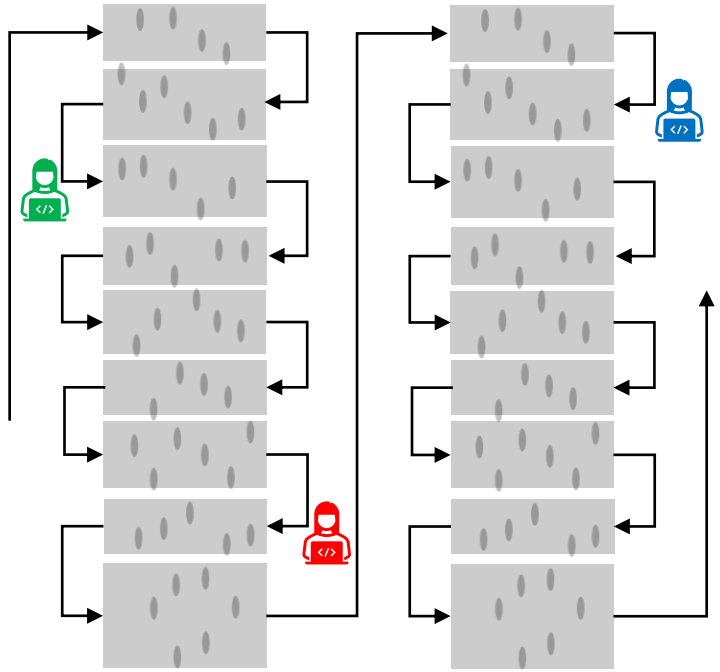
An adaptive parallelization protocol

- Need-based preregistration of updates
- Updates grouped into tasks
- Tasks linked within dynamic chain



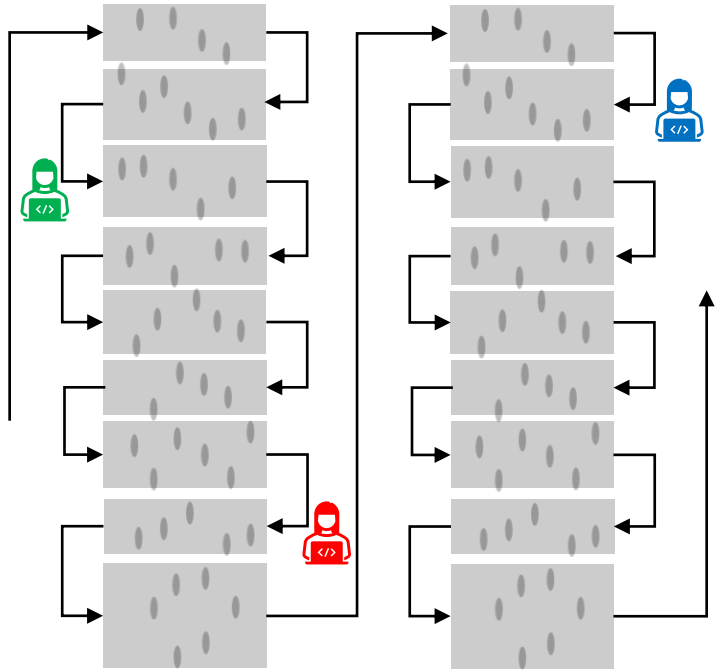
An adaptive parallelization protocol

- Need-based preregistration of updates
- Updates grouped into tasks
- Tasks linked within dynamic chain
- Several workers (equal roles) operating on chain
- Gracefully handling dependence relations



An adaptive parallelization protocol

- Need-based preregistration of updates
- Updates grouped into tasks
- Tasks linked within dynamic chain
- Several workers (equal roles) operating on chain
- Gracefully handling dependence relations
- Shared-memory paradigm
- Relatively easy to plug in any model



Experimental results

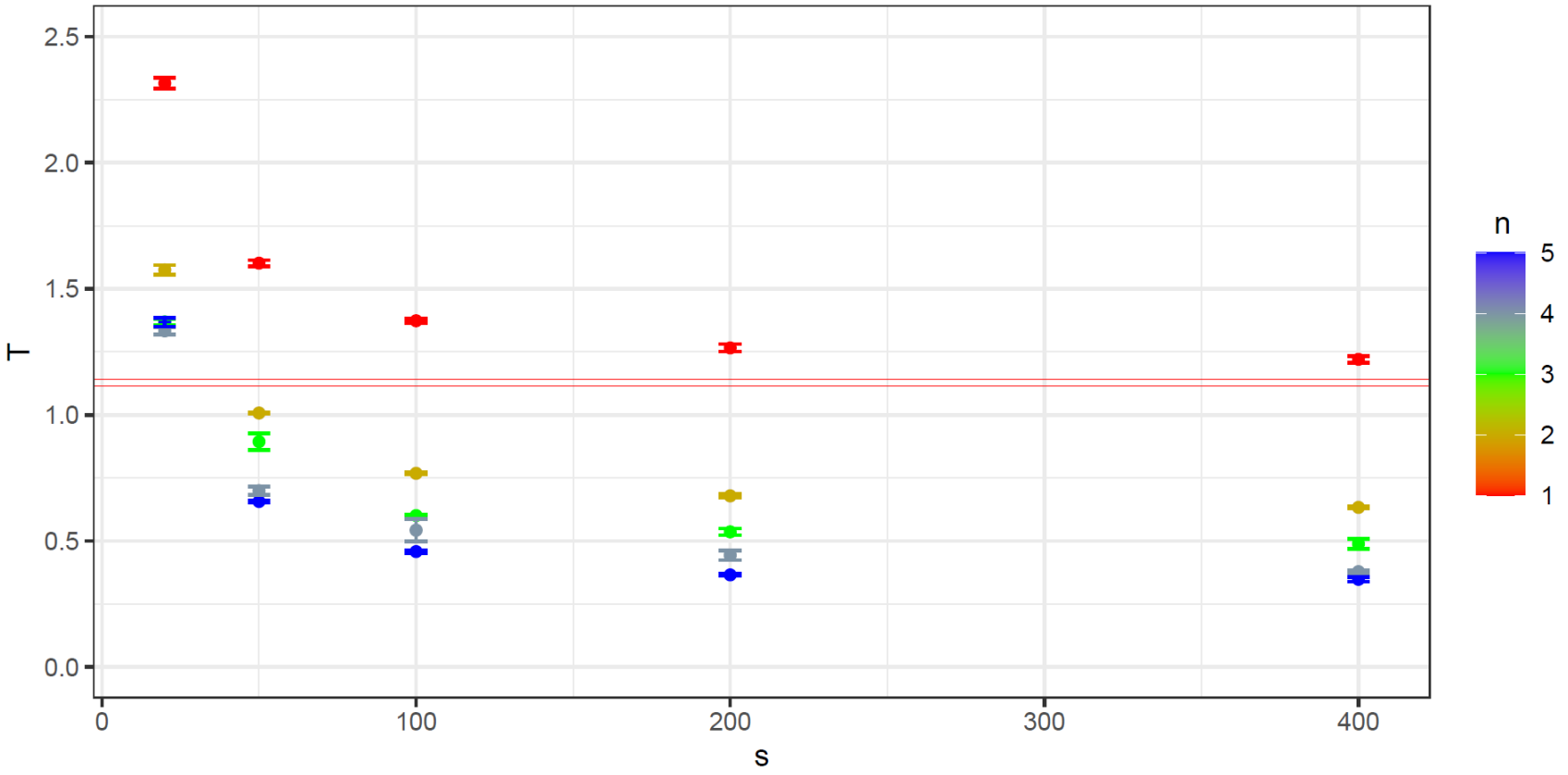
Model specifications:

- SIR-type disease spreading
- 4000 agents
- Ring-like network
- 14 links per node
- 3000x2 steps
- 2 types of agent updates:
  - Computing new state
  - Advancing to new state

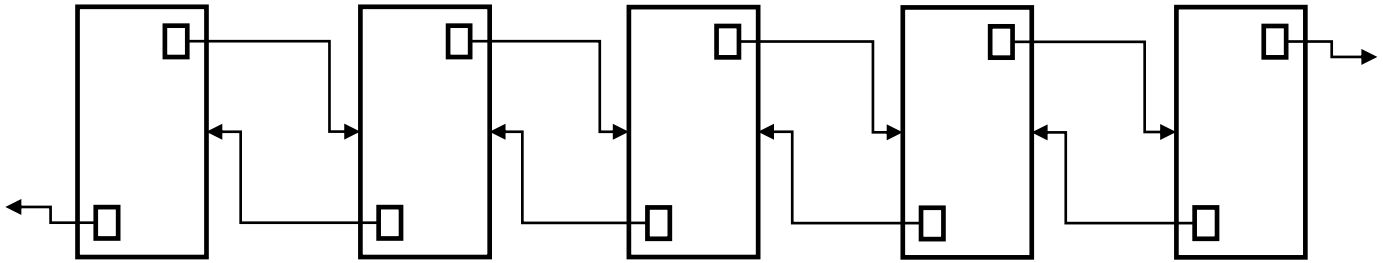
Experimental results

Model specifications:

- SIR-type disease spreading
- 4000 agents
- Ring-like network
- 14 links per node
- 3000x2 steps
- 2 types of agent updates:
  - Computing new state
  - Advancing to new state

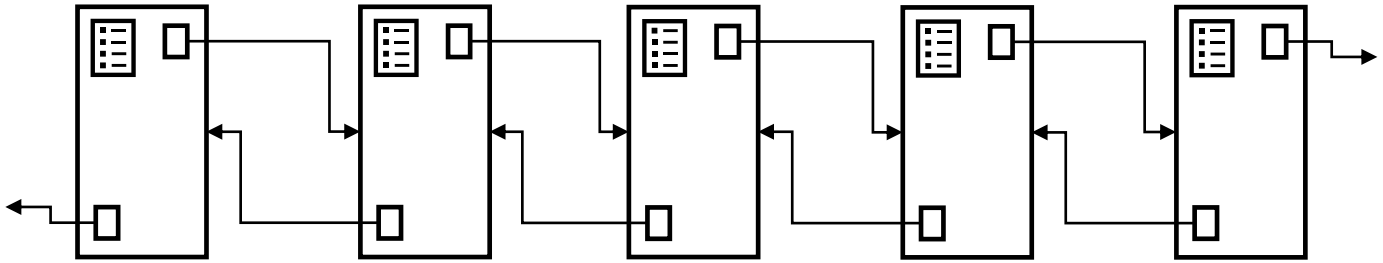


Protocol implementation



workers

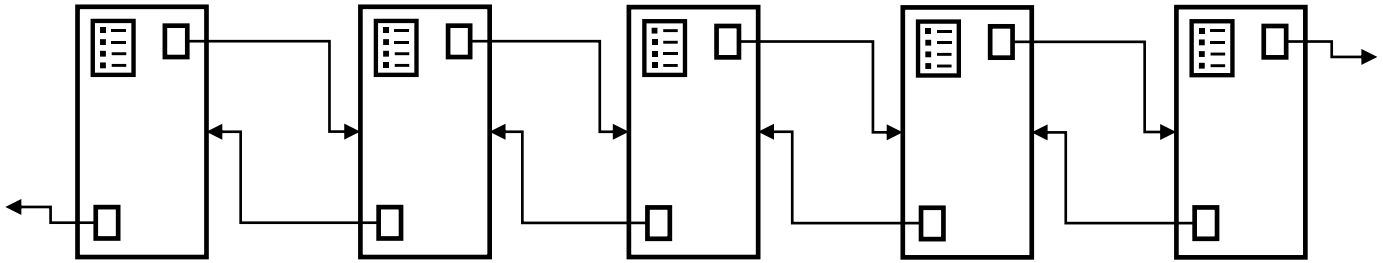
Protocol implementation



task **recipes**

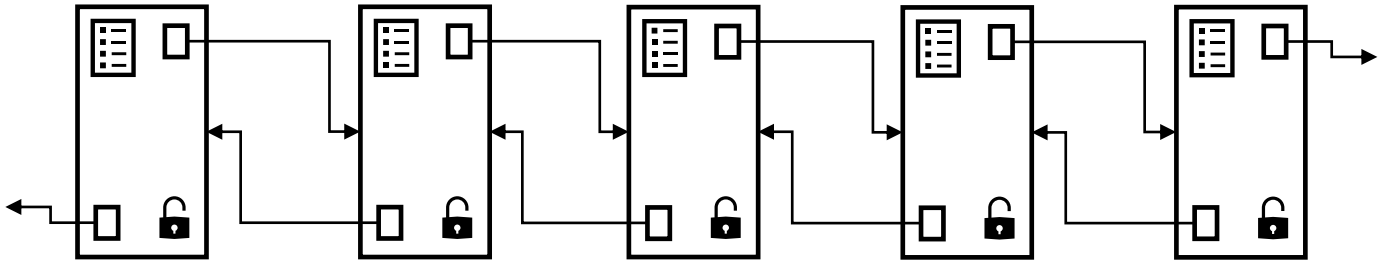
workers

Protocol implementation



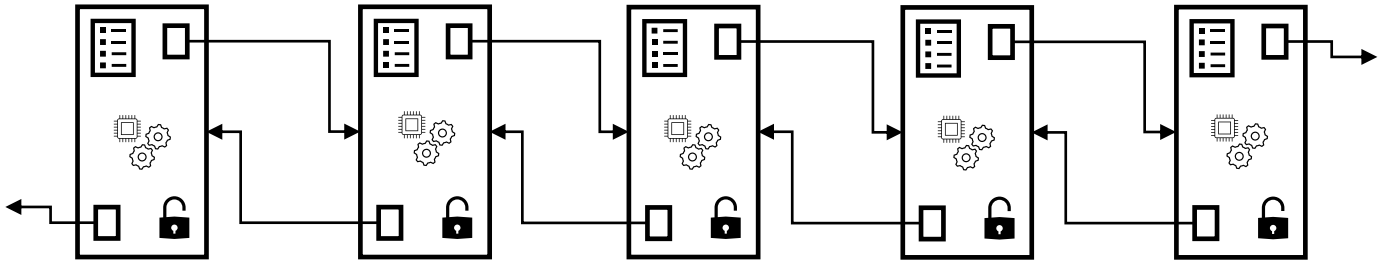
worker **records**

Protocol implementation



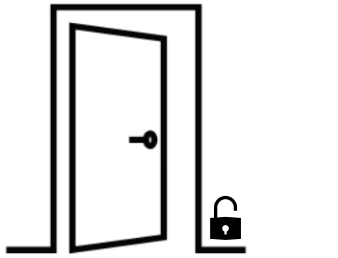
task mutex **locks**

Protocol implementation

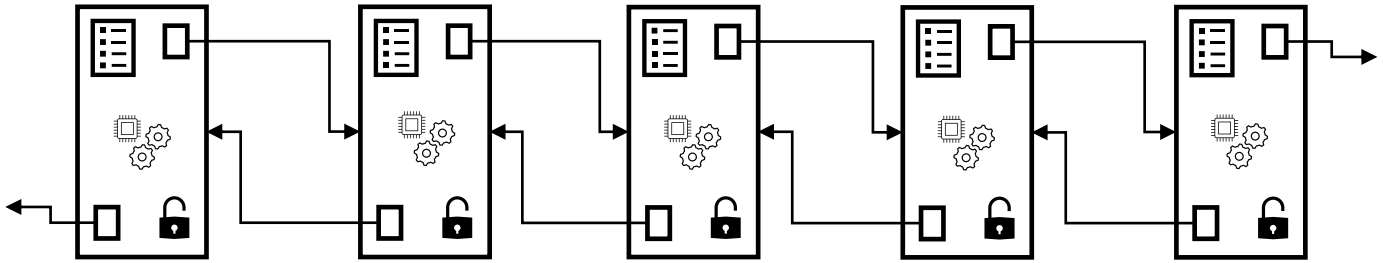


task execution **flags**

Protocol implementation



entering procedure

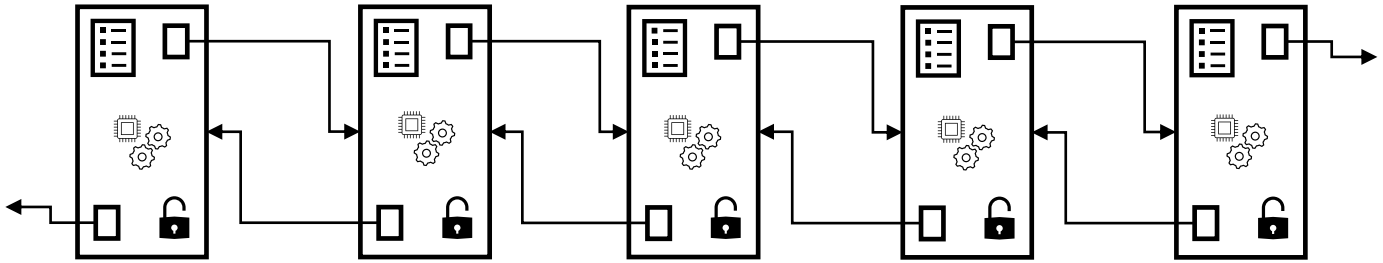


tasks

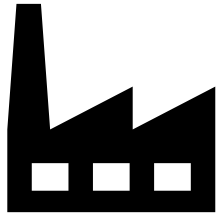
workers

Protocol implementation

entering procedure



tasks

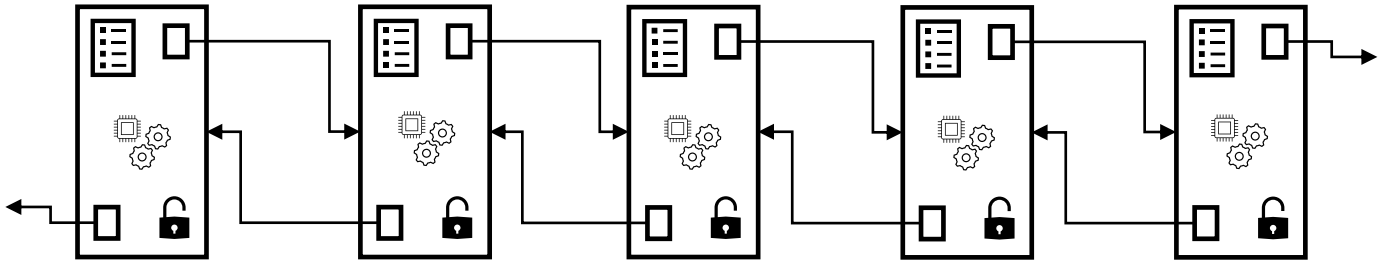


task creation procedure

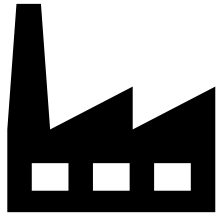
workers

Protocol implementation

entering procedure



tasks



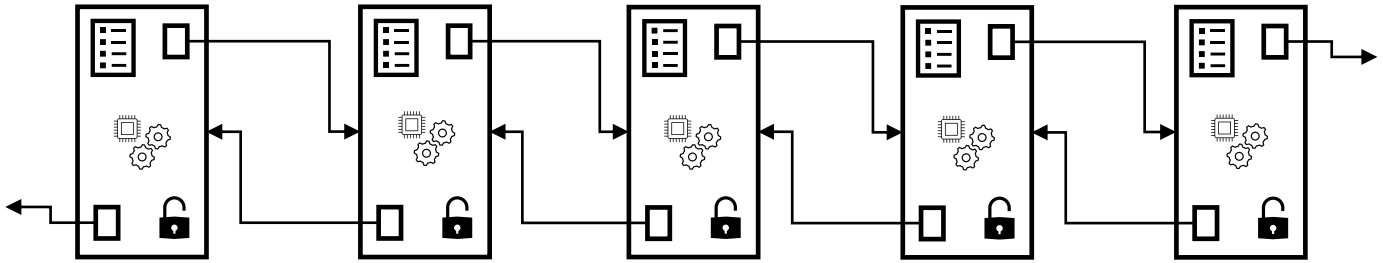
task creation procedure

task erasing procedure

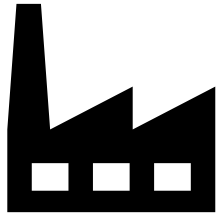
workers

Protocol implementation

entering procedure



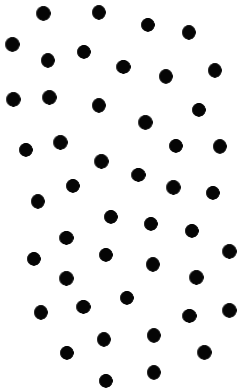
tasks



task creation procedure

task erasing procedure

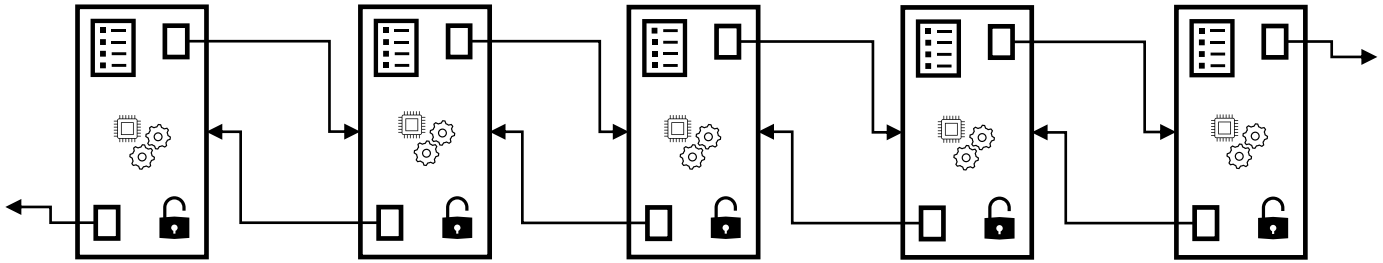
workers



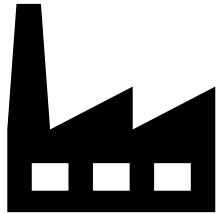
MABS model current state

Protocol implementation

entering procedure



tasks

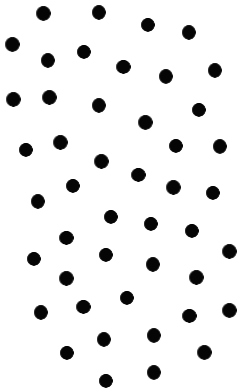


task creation procedure

task erasing procedure

workers

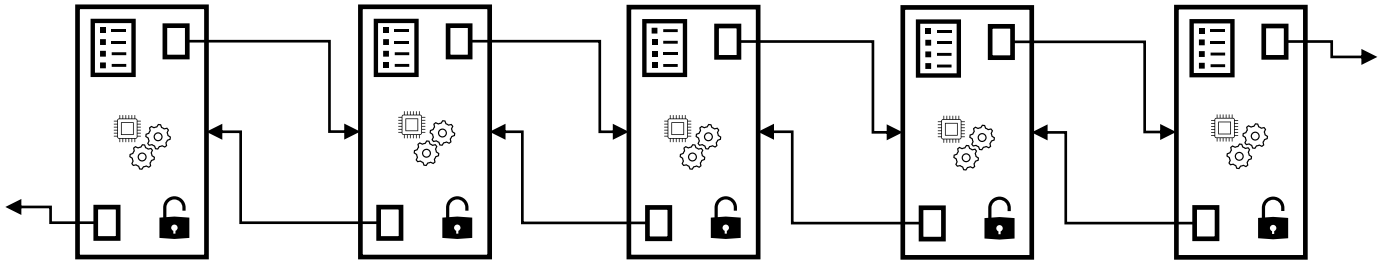
researcher



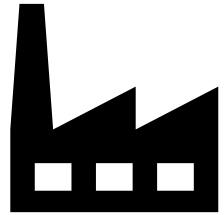
MABS model current state

Protocol implementation

entering procedure

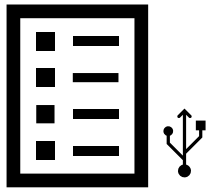


tasks



task creation procedure

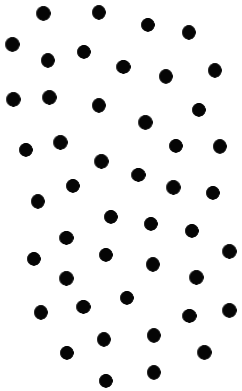
task erasing procedure



recipe class

record class

researcher



MABS model current state

workers

Conclusion and outlook

- Adaptive parallelism is a promising way to accelerate MABS

Conclusion and outlook

- Adaptive parallelism is a promising way to accelerate MABS
- It behaves as intended:
  - Adding workers reduces simulation time
  - If task size high enough
  - If not too many workers already

Conclusion and outlook

- Adaptive parallelism is a promising way to accelerate MABS
- It behaves as intended:
  - Adding workers reduces simulation time
  - If task size high enough
  - If not too many workers already
- More agents allow for larger tasks, hence more workers

Conclusion and outlook

- Adaptive parallelism is a promising way to accelerate MABS
- It behaves as intended:
  - Adding workers reduces simulation time
  - If task size high enough
  - If not too many workers already
- More agents allow for larger tasks, hence more workers
- Some upcoming work:
  - More detailed performance evaluation (benchmarks and variants)
  - Applying protocol to a more complicated MABS model (subsequent project)

Conclusion and outlook

- Adaptive parallelism is a promising way to accelerate MABS
- It behaves as intended:
  - Adding workers reduces simulation time
  - If task size high enough
  - If not too many workers already
- More agents allow for larger tasks, hence more workers
- Some upcoming work:
  - More detailed performance evaluation (benchmarks and variants)
  - Applying protocol to a more complicated MABS model (subsequent project)
- Workshop-version article (also) available online: arXiv:2304.01724

Backup

More experimental results