Uniform Bipartition in the Population Protocol Model with Arbitrary Communication Graphs

Hiroto Yasumi
Nara Institute of Science and Technology, Japan
yasumi.hiroto.yf9@is.naist.jp

Fukuhito Ooshita
Nara Institute of Science and Technology, Japan
f-oosita@is.naist.jp

Michiko Inoue
Nara Institute of Science and Technology, Japan
kounoe@is.naist.jp

Sébastien Tixeuil
Sorbonne Université, CNRS, LIP6, Paris, France
Sebastien.Tixeuil@lip6.fr

Abstract

In this paper, we focus on the uniform bipartition problem in the population protocol model. This problem aims to divide a population into two groups of equal size. In particular, we consider the problem in the context of *arbitrary* communication graphs. As a result, we investigate the solvability of the uniform bipartition problem with arbitrary communication graphs when agents in the population have designated initial states, under various assumptions such as the existence of a base station, symmetry of the protocol, and fairness of the execution. When the problem is solvable, we present protocols for uniform bipartition. When global fairness is assumed, the space complexity of our solutions is tight.

2012 ACM Subject Classification Theory of computation → Distributed algorithms; Theory of computation → Concurrent algorithms

Keywords and phrases population protocol, uniform bipartition, distributed protocol

Digital Object Identifier 10.4230/LIPIcs.OPODIS.2020.33

Related Version A full version of the paper is available at [34], `https://arxiv.org/abs/2011.08366`.

Funding This work was supported by JSPS KAKENHI Grant Numbers 18K11167, 20H04140, 20J21849, and JST SICORP Grant Number JPMJSC1806. This work was partially funded by the ANR project SAPPORO, ref. 2019-CE25-0005-1.

1 Introduction

1.1 Background

In this paper, we consider the population protocol model introduced by Angluin et al. [5]. The population protocol model is an abstract model for low-performance devices. In the population protocol model, devices are represented as anonymous agents, and a population is represented as a set of agents. Those agents move passively (*i.e.*, they cannot control their movements), and when two agents approach, they are able to communicate and update their states (this pairwise communication is called an interaction). A computation then consists of an infinite sequence of interactions.

24th International Conference on Principles of Distributed Systems (OPODIS 2020).
Editors: Quentin Bramas, Rotem Oshman, and Paolo Romano; Article No. 33; pp. 33:1–33:16
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Application domains for population protocols include sensor networks used to monitor live animals (each sensor is attached to a small animal and monitors *e.g.* its body temperature) that move unpredictably (hence, each sensor must handle passive mobility patterns). Another application domain is that of molecular robot networks [28]. In such systems, a large number of molecular robots collectively work inside a human body to achieve goals such as transport of medicine. Since those robots are tiny, their movement is uncontrollable, and robots may only maintain extremely small memory.

In the population protocol model, many researchers have studied various fundamental problems such as leader election protocols [4] (A population protocol solves leader election if starting from an initially uniform population of agents, eventually a single agent outputs *leader*, while all others output *non-leader*), counting [8, 10, 11] (The counting problem consists in counting how many agents participate to the protocol; As the agents' memory is typically constant, this number is output by a special agent that may maintain logarithmic size memory, the base station), majority [6] (The majority problem aims to decide which, if any, initial state in a population is a majority), k-partition [32, 35, 36] (The k-partition problem consists in dividing a population into k groups of equal size), etc.

In this paper, we focus on the uniform bipartition problem [33, 35, 36], whose goal is to divide a population into two stable groups of equal size (the difference is one if the population size is odd). To guarantee the stability of the group, each agent eventually belongs to a single group and never changes the group after that. Applications of the uniform bipartition include saving batteries in a sensor network by switching on only one group, or executing two tasks simultaneously by assigning one task to each group. Contrary to previous work that considered *complete* communication graphs [33, 36], we consider the uniform bipartition problem over *arbitrary* graphs. In the population protocol model, most existing works consider the complete communication graph model (every pairwise interaction is feasible). However, realistic networks command studying incomplete communication graphs (where only a subset of pairwise interactions remains feasible) as low-performance devices and unpredictable movements may not yield a complete set of interactions. Moreover, in this paper, we assume the designated initial states (i.e., all agents share the same given initial state), and consider the problem under various assumptions such as the existence of a base station, symmetry of the protocol, and fairness of the execution. Although protocols with arbitrary initial states tolerate a transient fault, protocols with designated initial states can usually be designed using fewer states, and exhibit faster convergence times. Actually, it was shown in [35] that, with arbitrary initial states, constant-space protocols cannot be constructed in most cases even assuming complete graphs.

1.2 Related Works

The population protocol model was proposed by Angluin et al. [5], who were recently awarded the 2020 Edsger W. Dijkstra prize in Distributed Computing for their work. While the core of the initial study was dedicated to the computability of the model, subsequent works considered various problems (*e.g.*, leader election, counting, majority, uniform k-partition) under different assumptions (*e.g.*, existence of a base station, fairness, symmetry of protocols, and initial states of agents).

The *leader election* problem was studied from the perspective of time and space efficiency. Doty and Soloveichik [20] proved that $\Omega(n)$ expected parallel time is required to solve leader election with probability 1 if agents have a constant number of states. Relaxing the number of states to a polylogarithmic value, Alistarh and Gelashvili [3] proposed a leader election protocol in polylogarithmic expected stabilization time. Then, Gąsieniec et al. [23] designed

Table 1 The minimum number of states to solve the uniform bipartition problem with designated initial states over *complete* graphs [33, 35].

base station	fairness	symmetry	upper bound	lower bound
initialized/non-initialized base station	global	asymmetric	3	3
		symmetric	3	3
	weak	asymmetric	3	3
		symmetric	3	3
no base station	global	asymmetric	3	3
		symmetric	4	4
	weak	asymmetric	3	3
		symmetric	unsolvable	

a protocol with $O(\log \log n)$ states and $O(\log n \cdot \log \log n)$ expected time. Furthermore, the protocol of Gąsieniec et al. [23] is space-optimal for solving the problem in polylogarithmic time. In [29], Sudo et al. presented a leader election protocol with $O(\log n)$ states and $O(\log n)$ expected time. This protocol is time-optimal for solving the problem. Finally, Berenbrink et al. [15] proposed a time and space optimal protocol that solves the leader election problem with $O(\log \log n)$ states and $O(\log n)$ expected time. In the case of arbitrary communication graphs, it turns out that self-stabilizing leader election is impossible [7] (a protocol is self-stabilizing if its correctness does not depend on its initial global state). This impossibility can be avoided if oracles are available [9, 18] or if the self-stabilization requirement is relaxed: Sudo et al. [30] proposed a loosely stabilizing protocol for leader election (loose stabilization relates to the fact that correctness is only guaranteed for a very long expected amount of time).

The *counting* problem was introduced by Beauquier et al. [11] and popularized the concept of a base station. Space complexity was further reduced by follow-up works [10, 24], until Aspnes et al. [8] finally proposed a time and space optimal protocol. On the other hand, by allowing the initialization of agents, the counting protocols without the base station were proposed for both exact counting [16] and approximate counting [1, 16]. In [1], Alistarh et al. proposed a protocol that computes an integer k such that $\frac{1}{2} \log n < k < 9 \log n$ in $O(\log n)$ time with high probability using $O(\log n)$ states. After that, Berenbrink et al. [16] designed a protocol that outputs either $\lfloor \log n \rfloor$ or $\lceil \log n \rceil$ in $O(\log^2 n)$ time with high probability using $O(\log n \cdot \log \log n)$ states. Moreover, in [16], they proposed the exact counting protocol that computes n in $O(\log n)$ time using $\tilde{O}(n)$ states with high probability.

The *majority* problem was addressed under different assumptions (*e.g.*, with or without failures [6], deterministic [22, 25] or probabilistic [2, 12, 13, 25] solutions, with arbitrary communication graphs [27], etc.). Those works also consider minimizing the time and space complexity. Berenbrink et al. [14] show trade-offs between time and space for the problem.

To our knowledge, the *uniform k-partition* problem and its variants have only been considered in complete communication graphs. Lamani et al. [26] studied a group decomposition problem that aims to divide a population into groups of designated sizes. Yasumi et al. [32] proposed a uniform k-partition protocol with no base station. Umino et al. [31] extended the result to the R-generalized partition problem that aims at dividing a population into k groups whose sizes follow a given ratio R. Also, Delporte-Gallet et al. [19] proposed a k-partition protocol with relaxed uniformity constraints: the population is divided into k groups such that in any group, at least $n/(2k)$ agents exist, where n is the number of agents.

Most related to our work is the uniform bipartition solution for complete communication graphs provided by Yasumi et al. [33, 35]. For the uniform bipartition problem over complete graphs with designated initial states, Yasumi et al. [33, 35] studied space complexity under

Table 2 The minimum number of states to solve the uniform bipartition problem with designated initial states over *arbitrary* graphs. P is a known upper bound of the number of agents, and $l \geq 3$ and h are positive integers.

base station	fairness	symmetry	upper bound	lower bound
initialized/non-initialized base station	global	asymmetric	3*	$3^\dagger$
		symmetric	3*	$3^\dagger$
	weak	asymmetric	$3P + 1$* $3l + 1$ for no $l \cdot h$ cycle *	$3^\dagger$
		symmetric	$3P + 1$* $3l + 1$ for no $l \cdot h$ cycle *	$3^\dagger$
no base station	global	asymmetric	4*	4*
		symmetric	5*	5*
	weak	asymmetric	unsolvable*	
		symmetric	unsolvable†	

* Contributions of this paper

† Deduced from Yasumi et al. [35]

various assumptions such as: *(i)* an initialized base station, a non-initialized base station, or no base station (an initialized base station has a designated initial state, while a non-initialized has an arbitrary initial state), *(ii)* asymmetric or symmetric protocols (asymmetric protocols allow interactions between two agents with the same state to map to two resulting different states, while symmetric protocols do not allow such a behavior), and *(iii)* global or weak fairness (weak fairness guarantees that every individual pairwise interaction occurs infinitely often, while global fairness guarantees that every recurrently reachable configuration is eventually reached). Furthermore, they also study the solvability of the uniform bipartition problem with arbitrary initial states. Table 1 shows the minimum number of states to solve the uniform bipartition with designated initial states over complete communication graphs.

There exist some protocol transformers that transform protocols for some assumptions into ones for other assumptions. In [5], Angluin et al. proposed a transformer that transforms a protocol with complete communication graphs into a protocol with arbitrary communication graphs. This transformer requires the quadruple state space and works under global fairness. In this transformer, agents exchange their states even after convergence. For the uniform bipartition problem, since agents must keep their groups after convergence, they cannot exchange their states among different groups and thus the transformer proposed in [5] cannot directly apply to the uniform bipartition problem. Bournez et al. [17] proposed a transformer that transforms an asymmetric protocol into symmetric protocol by assuming additional states. In [17], only protocols with complete communication graphs were considered and the transformer works under global fairness. We use the same idea to construct a symmetric uniform bipartition protocol under global fairness without a base station.

1.3 Our Contributions

In this paper, we study the solvability of the uniform bipartition problem with designated initial states over arbitrary graphs. A summary of our results is presented in Table 2. Let us first observe that, as complete communication graphs are a special case of arbitrary communication graphs, the impossibility results by Yasumi et al. [35] remain valid in our setting. With a base station (be it initialized or non-initialized) under global fairness, we extend the three states protocol by Yasumi et al. [35] from complete communication graphs to arbitrary communication graphs. With a non-initialized base station under weak fairness, we propose a new symmetric protocol with $3P + 1$ states, where P is a known upper bound of the number of agents. These results yield identical upper bounds for the easier cases of asymmetric protocols and/or initialized base station. In addition, we also show a condition of communication graphs in which the number of states in the protocol can be reduced from $3P + 1$ to constant. Concretely, we show that the number of states in the protocol can be reduced to $3l + 1$ if we assume communication graphs such that every cycle either includes the base station or its length is not a multiple of l, where l is a positive integer at least three. On the other hand, with no base station under global fairness, we prove that four and five states are necessary and sufficient to solve uniform bipartition with asymmetric and symmetric protocols, respectively. In the same setting, in complete graphs, three and four states were necessary and sufficient. So, one additional state enables problem solvability in arbitrary communications graphs in this setting. With no base station under weak fairness, we prove that the problem cannot be solved, using a similar argument as in the impossibility result for leader election by Fischer and Jiang [21]. Overall, we show the solvability of uniform bipartition in a variety of settings for a population of agents with designated initial states assuming arbitrary communication graphs. In cases where the problem remains feasible, we provide upper and lower bounds with respect to the number of states each agent maintains, and in all cases where global fairness can be assumed, our bounds are tight.

In this paper, because of space limitations, we omitted proofs of lemmas and theorems (see the full version [34]).

2 Definitions

2.1 Population Protocol Model

A population whose communication graph is arbitrary is represented by an undirected connected graph $G = (V, E)$, where V is a set of agents, and $E \subseteq V \times V$ is a set of edges that represent the possibility of an interaction between two agents. That is, two agents $u \in V$ and $v \in V$ can interact only if $(u, v) \in E$ holds. A protocol $\mathcal{P} = (Q, \delta)$ consists of Q and δ, where Q is a set of possible states for agents, and δ is a set of transitions from $Q \times Q$ to $Q \times Q$. Each transition in δ is denoted by $(p, q) \to (p', q')$, which means that, when an interaction between an agent x in state p and an agent y in state q occurs, their states become p' and q', respectively. Moreover, we say x is an initiator and y is a responder. When x and y interact as an initiator and a responder, respectively, we simply say that x interacts with y. Transition $(p, q) \to (p', q')$ is null if both $p = p'$ and $q = q'$ hold. We omit null transitions in the descriptions of protocols. Protocol $\mathcal{P} = (Q, \delta)$ is symmetric if, for every transition $(p, q) \to (p', q')$ in δ, $(q, p) \to (q', p')$ exists in δ. In particular, if a protocol $\mathcal{P} = (Q, \delta)$ is symmetric and transition $(p, p) \to (p', q')$ exists in δ, $p' = q'$ holds. If a protocol is not symmetric, the protocol is asymmetric. Protocol $\mathcal{P} = (Q, \delta)$ is deterministic if, for any pair of states $(p, q) \in Q \times Q$, exactly one transition $(p, q) \to (p', q')$ exists in δ. We consider only deterministic protocols in this paper. A global state of a population is called a configuration, defined as a vector of (local) states of all agents. A state of agent a in configuration C, is denoted by $s(a, C)$. Moreover, when C is clear from the context, we simply use $s(a)$ to denote the state of agent a. A transition between two configurations C and C' is described as $C \to C'$, and means that configuration C' is obtained from C by a single interaction between two agents. For two configurations C and C', if there exists a sequence of configurations $C = C_0, C_1, \dots, C_m = C'$ such that $C_i \to C_{i+1}$ holds for every i $(0 \leq i < m)$, we say C' is reachable from C, denoted by $C \xrightarrow{*} C'$. An infinite sequence of configurations $\Xi = C_0, C_1, C_2, \dots$ is an execution of a protocol if $C_i \to C_{i+1}$ holds for every i ($i \geq 0$). An execution Ξ is weakly-fair if, for each pair of agents $(v, v') \in E$, v (resp. v') interacts with v' (resp., v) infinitely often[1]. An execution Ξ is globally-fair if, for every pair of configurations C and C' such that $C \to C'$, C' occurs infinitely often when C occurs infinitely often. Intuitively, global fairness guarantees that, if configuration C occurs infinitely often, then every possible interaction in C also occurs infinitely often. Then, if C occurs infinitely often, C' satisfying $C \to C'$ occurs infinitely often, we can deduce that C'' satisfying $C' \to C''$ also occurs infinitely often. Overall, with global fairness, if a configuration C occurs infinitely often, then every configuration C^* reachable from C also occurs infinitely often.

In this paper, we consider three possibilities for the base station: initialized base station, non-initialized base station, and no base station. In the model with a base station, we assume that a single agent, called a base station, exists in V. Then, V can be partitioned into V_b, the singleton set containing the base station, and V_p, the set of agents except for the base station. The base station can be distinguished from other agents in V_p, although agents in V_p cannot be distinguished. Then, the state set Q can be partitioned into a state set Q_b for the base station, and a state set Q_p for agents in V_p. The base station has unlimited resources (with respect to the number of states), in contrast with other resource-limited agents (that are allowed only a limited number of states). So, when we evaluate the space complexity of a protocol, we focus on the number of states $|Q_p|$ for agents in V_p and do not consider the number of states $|Q_b|$ that are allocated to the base station. In the sequel, we thus say a protocol uses x states if $|Q_p| = x$ holds. When we assume an initialized base station, the base station has a designated initial state. When we assume a non-initialized base station, the base station has an arbitrary initial state (in Q_b), although agents in V_p have the same designated initial state. When we assume no base station, there exists no base station and thus $V = V_p$ holds. For simplicity, we use agents only to refer to agents in V_p in the following sections. To refer to the base station, we always use the term base station (not an agent). In the initial configuration, both the base station and the agents are not aware of the number of agents, yet they are given an upper bound P of the number of agents. However, in protocols except for a protocol in Section 3.2, we assume that they are not given P.

2.2 Uniform Bipartition Problem

Let $f : Q_p \to \{red, blue\}$ be a function that maps a state of an agent to *red* or *blue*. We define the color of an agent a as $f(s(a))$. Then, we say that agent a is *red* (resp., *blue*) if $f(s(a)) = red$ (resp., $f(s(a)) = blue$) holds. If an agent a has state s such that $f(s) = red$ (resp., $f(s) = blue$), we call a a *red* agent (resp., a *blue* agent). For some population V, the number of *red* agents (resp., *blue* agents) in V is denoted by $\#red(V)$ (resp., $\#blue(V)$). When V is clear from the context, we simply write $\#red$ and $\#blue$.

A configuration C is stable with respect to the uniform bipartition if there exists a partition $\{H_r, H_b\}$ of V_p that satisfies the following conditions:

1. $||H_r| - |H_b|| \leq 1$ holds, and
2. For every configuration C' such that $C \xrightarrow{*} C'$, each agent in H_r (resp., H_b) remains *red* (resp., *blue*) in C'.

---

[1] We use this definition for the lower bound under weak fairness, but for the upper bound we use a slightly weaker version. We show that our proposed protocols for weak fairness works if, for each pair of agents $(v, v') \in E$, v and v' interact infinitely often (i.e., for interactions by some pair of agents v and v', it is possible that v only becomes an initiator and v' never becomes an initiator).

An execution $\Xi = C_0, C_1, C_2, \ldots$ solves the uniform bipartition problem if Ξ includes a configuration C_t that is stable for uniform bipartition. Finally, a protocol $\mathcal{P}$ solves the uniform bipartition problem if every possible execution Ξ of protocol $\mathcal{P}$ solves the uniform bipartition problem.

3 Upper Bounds with a Non-initialized Base Station

In this section, we prove some upper bounds on the number of states that are required to solve the uniform bipartition problem over arbitrary graphs with designated initial states and a non-initialized base station. More concretely, with global fairness, we propose a symmetric protocol with three states by extending the protocol by Yasumi et al. [35] from a complete communication graph to an arbitrary communication graph. In the case of weak fairness, we present a symmetric protocol with $3P + 1$ states, where P is a known upper bound of the number of agents.

3.1 Upper Bound for Symmetric Protocols under Global Fairness

The state set of agents in this protocol is $Q_p = \{initial, red, blue\}$, and we assume that $f(initial) = f(red) = red$ and $f(blue) = blue$ hold. The designated initial state of agents is *initial*. The idea of the protocol is as follows: the base station assigns *red* and *blue* to agents whose state is *initial* alternately. As the base station cannot meet every agent (the communication graph is arbitrary), the positions of state *initial* are moved throughout the communication graph using transitions. Thus, if an agent with *initial* state exists somewhere in the network, the base station has infinitely many chances to interact with a neighboring agent with *initial* state. This implies that the base station is able to repeatedly assign *red* and *blue* to neighboring agents with *initial* state unless no agent anywhere in the network has *initial* state. Since the base station assigns *red* and *blue* alternately, the uniform bipartition is completed after no agent has *initial* state.

To make *red* and *blue* alternately, the base station has a state set $Q_b = \{b_{red}, b_{blue}\}$. Using its current state, the base station decides which color to use for the next interaction with a neighboring agent with *initial* state. Now, to move the position of an *initial* state in the communication graph, if an agent with *initial* state and an agent with *red* (or *blue*) state interact, they exchange their states. This implies that eventually an agent adjacent to the base station has *initial* state and then the agent and the base station interact (global fairness guarantees that such interaction eventually happens). Transition rules of the protocol are the following (for each transition rule $(p, q) \to (p', q')$, transition rule $(q, p) \to (q', p')$ exists, but we omit the description).

1. $(b_{red}, initial) \to (b_{blue}, red)$
2. $(b_{blue}, initial) \to (b_{red}, blue)$
3. $(blue, initial) \to (initial, blue)$
4. $(red, initial) \to (initial, red)$

From these transition rules, the protocol converges when no agent has *initial* state (indeed, no interaction is defined when no agent has *initial* state).

▶ **Theorem 1.** *In the population protocol model with a non-initialized base station, there exists a symmetric protocol with three states per agent that solves the uniform bipartition problem with designated initial states assuming global fairness in arbitrary communication graphs.*

Algorithm 1 Uniform bipartition protocol with $3P+1$ states.

```
Variables at the base station:
    $RB \in \{r, b\}$: The state that the base station assigns next
Variables at an agent $x$:
    $color_x \in \{ini, r, b\}$: Color of the agent, initialized to $ini$
    $depth_x \in \{\perp, 1, 2, 3, \dots, P\}$: Depth of agent $x$ in a tree rooted at the base station,
    initialized to $\perp$
when an agent $x$ and the base station interact do
    if $color_x = ini$ and $depth_x = 1$ then
        $color_x \leftarrow RB$
        $RB \leftarrow \overline{RB}$
    if $depth_x = \perp$ then $depth_x \leftarrow 1$
when two agents $x$ and $y$ interact do
    if $depth_y \neq \perp$ and $depth_x = \perp$ then $depth_x \leftarrow depth_y + 1$
    else if $depth_x \neq \perp$ and $depth_y = \perp$ then $depth_y \leftarrow depth_x + 1$
    if $depth_x < depth_y$ and $color_y = ini$ then
        $color_y \leftarrow color_x$
        $color_x \leftarrow ini$
    if $depth_y < depth_x$ and $color_x = ini$ then
        $color_x \leftarrow color_y$
        $color_y \leftarrow ini$
Note: If $depth_x = \perp$ holds, $color_x = ini$ holds.
```

Note that, under weak fairness, this protocol does not solve the uniform bipartition problem. This is because we can construct a weakly-fair execution of this protocol such that some agents keep *initial* state infinitely often. For example, we can make an agent keep *initial* by constructing an execution in the following way.

- If the agent (in *initial*) interacts with an agent in *red* or *blue*, the next interaction occurs between the same pair of agents.

3.2 Upper Bound for Symmetric Protocols under Weak Fairness

3.2.1 A protocol over arbitrary graphs

In this protocol, every agent x has variables $color_x$ and $depth_x$. Variable $color_x$ represents the color of agent x. That is, for an agent x, if $color_x = ini$ or $color_x = r$ holds, $f(s(x)) = red$ holds. On the other hand, if $color_x = b$ holds, $f(s(x)) = blue$ holds. The protocol is given in Algorithm 1. Note that this algorithm does not care an initiator and a responder.

The basic strategy of the protocol is the following.

1. Create a spanning tree rooted at the base station. Concretely, agent x assigns its depth in a tree rooted at the base station into variable $depth_x$. Variable $depth_x$ is initialized to $\perp$. Variable $depth_x$ obtains the depth of x in the spanning tree as follows: If the base station and an agent p with $depth_p = \perp$ interact, $depth_p$ becomes 1. If an agent q with $depth_q \neq \perp$ and an agent p with $depth_p = \perp$ interact, $depth_p$ becomes $depth_q + 1$. By these behaviors, for any agent x, eventually variable $depth_x$ has a depth of x in a tree rooted at the base station.
2. Using the spanning tree, carry the initial color ini toward the base station and make the base station assign r and b to agents one by one. Concretely, if agents x and y interact

and both $depth_y < depth_x$ and $color_x = ini$ hold, x and y exchange their colors (i.e., ini is carried from x to y). Hence, since ini is always carried to a smaller $depth$, eventually an agent z with $depth_z = 1$ obtains ini. After that, the base station and the agent z interact and the base station assigns r or b to z. Note that, if the base station assigns r (resp., b), the base station assigns b (resp., r) next.

Then, for any agent v, eventually $color_v \neq ini$ holds. Hence, there exist $\lceil n/2 \rceil$ *red* (resp., *blue*) agents, and $\lfloor n/2 \rfloor$ *blue* (resp., *red*) agents if variable RB in the base station has r (resp., b) as an initial value. Therefore, the protocol solves the uniform bipartition problem.

▶ **Theorem 2.** *Algorithm 1 solves the uniform bipartition problem. That is, there exists a protocol with* $3P + 1$ *states and designated initial states that solves the uniform bipartition problem under weak fairness assuming arbitrary communication graphs with a non-initialized base station.*

3.2.2 A protocol with constant states over a restricted class of graphs

In this subsection, we show that the space complexity of Algorithm 1 can be reduced to constant for communication graphs such that every cycle either includes the base station or its length is not a multiple of l, where l is a positive integer at least three.

We modify Algorithm 1 as follows. Each agent maintains the distance from the base station by computing modulo l plus 1. That is, we change lines 7 and 8 in Algorithm 1 to $depth_x \leftarrow depth_y \mod l + 1$ and $depth_y \leftarrow depth_x \mod l + 1$, respectively. Now $depth_x \in \{\bot, 1, 2, 3, \ldots, l\}$ holds for any agent x. Then we redefine the relation $depth_x < depth_y$ in lines 9 and 12 as follows: $depth_x < depth_y$ holds if and only if either $depth_x = 1 \land depth_y = 2$, $depth_x = 2 \land depth_y = 3$, $depth_x = 3 \land depth_y = 4$, …, $depth_x = l - 1 \land depth_y = l$, or $depth_x = l \land depth_y = 1$ holds.

We can easily observe that these modifications do not change the essence of Algorithm1. For two agents x and y, we say $x < y$ if $depth_x < depth_y$ holds. Each agent x eventually assigns a depth of x modulo l plus 1 to $depth_x$, and at that time there exists a path $x_0, x_1, \ldots, x_h$ such that x_0 is a neighbor of the base station, $x = x_h$ holds, and $x_i < x_{i+1}$ holds for any $0 \leq i < h$. In addition, there exists no cycle $x_0, x_1, \ldots, x_h = x_0$ such that $x_i < x_{i+1}$ holds for any $0 \leq i < h$. This is because, from the definition of relation '<', the length of such a cycle should be a multiple of l, but we assume that underlying communication graphs do not include a cycle of agents in V_p whose length is a multiple of l. Hence, similarly to Algorithm 1, we can carry the initial color ini toward the base station and make the base station assign r and b to agents one by one.

▶ **Corollary 3.** *There exists a protocol with* $3l + 1$ *states and designated initial states that solves the uniform bipartition problem under weak fairness assuming arbitrary communication graphs with a non-initialized base station if, for any cycle of the communication graphs, it either includes the base station or its length is not a multiple of* l*, where* l *is a positive integer at least three.*

4 Upper and Lower Bounds with No Base Station

In this section, we show upper and lower bounds of the number of states to solve the uniform bipartition problem with no base station and designated initial states over arbitrary communication graphs. Concretely, under global fairness, we prove that the minimum number of states for asymmetric protocols is four, and the minimum number of states for symmetric protocols is five. Under weak fairness, we prove that the uniform bipartition problem cannot be solved without a base station using proof techniques similar to those Fischer and Jiang [21] used to show the impossibility of leader election.

Algorithm 2 Transition rules of the uniform bipartition protocol with four states.

1. $(r^\omega, r^\omega) \to (r, b)$
2. $(r^\omega, b^\omega) \to (b, b)$
3. $(r^\omega, r) \to (r, r^\omega)$
4. $(b^\omega, b) \to (b, b^\omega)$
5. $(r^\omega, b) \to (r, b^\omega)$
6. $(b^\omega, r) \to (b, r^\omega)$

4.1 Upper Bound for Protocols under Global Fairness

In this subsection, over arbitrary graphs with designated initial states and no base station under global fairness, we give an asymmetric protocol with four states and a symmetric protocol with five states.

First, we show the asymmetric protocol with four states. We define a state set of agents as $Q = \{r^\omega, b^\omega, r, b\}$, and function f as follows: $f(r^\omega) = f(r) = red$ and $f(b^\omega) = f(b) = blue$. We say an agent has a token if its state is r^ω or b^ω. Initially, every agent has state r^ω, that is, every agent is *red* and has a token. The transition rules are given in Algorithm 2 (for each transition rule $(p, q) \to (p', q')$ except for transition rule 1, transition rule $(q, p) \to (q', p')$ exists, but we omit the description).

The basic strategy of the protocol is as follows. When two agents with tokens interact and one of them is *red*, a *red* agent transitions to *blue* and the two tokens are deleted (transition rules 1 and 2). Since n tokens exist initially and the number of tokens decreases by two in an interaction, $\lfloor n/2 \rfloor$ *blue* agents appear and $\lceil n/2 \rceil$ *red* agents remain after all tokens (except one token for the case of odd n) disappear. To make such interactions, the protocol moves a token when agents with and without a token interact (transition rules 3, 4, 5, and 6). Global fairness guarantees that, if two tokens exist, an interaction of transition rule 1 or 2 happens eventually. Therefore, the uniform bipartition is achieved by the protocol.

▶ **Theorem 4.** *Algorithm 2 solves the uniform bipartition problem. That is, there exists a protocol with four states and designated initial states that solves the uniform bipartition problem under global fairness over arbitrary communication graphs.*

Furthermore, we obtain a symmetric protocol under the assumption by using a similar idea of the transformer proposed in [17]. The transformer simulates an asymmetric protocol on a symmetric protocol. To do this, the transformer requires additional states. Moreover, the transformer works with complete communication graphs. We show that one additional state is sufficient to transform the asymmetric uniform bipartition protocol into the symmetric protocol even if we assume arbitrary graphs (see the full version [34]).

▶ **Theorem 5.** *There exists a symmetric protocol with five states and designated initial states that solves the uniform bipartition problem under global fairness with arbitrary communication graphs.*

4.2 Lower Bound for Asymmetric Protocols under Global Fairness

In this section, we show that, over arbitrary graphs with designated initial states and no base station under global fairness, there exists no asymmetric protocol with three states.

To prove this, we first show that, when the number of agents n is odd and no more than $P/2$, each agent changes its own state to another state infinitely often in any globally-fair execution Ξ of a uniform bipartition protocol Alg, where P is a known upper bound of the number of agents. This lemma holds regardless of the number of states in a protocol.

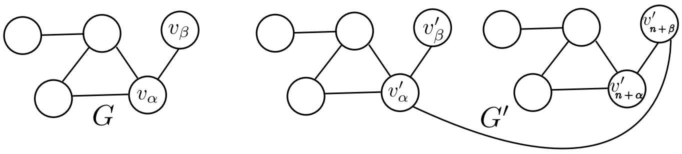

Figure 1 An example of communication graphs G and G' ($n = 5$).

▶ **Lemma 6.** *Assume that there exists a uniform bipartition protocol Alg with designated initial states over arbitrary communication graphs assuming global fairness. Consider a graph $G = (V, E)$ such that the number of agents n is odd and no more than $P/2$. In any globally-fair execution $\Xi = C_0, C_1, \ldots$ of Alg over G, each agent changes its state infinitely often.*

Proof. (Proof sketch) First, for the purpose of contradiction, we assume that there exists an agent v_α that never changes its state after some stable configuration C_h in a globally-fair execution Ξ over graph G. Let s_α be a state that v_α has after C_h. Let $v_\beta \in V$ be an agent adjacent to v_α and S_β be a set of states that v_β has after C_h. Since the number of states is finite, there exists a stable configuration C_t that occurs infinitely often after C_h. Next, let $G'_1 = (V'_1, E_1)$ and $G'_2 = (V'_2, E_2)$ be graphs that are isomorphic to G. Moreover, let $v'_\alpha \in V'_1$ (resp., $v'_{n+\beta} \in V'_2$) be an agent that corresponds to $v_\alpha \in V$ (resp., $v_\beta \in V$). We construct $G' = (V', E')$ by connecting G'_1 and G'_2 with an additional edge $(v'_\alpha, v'_{n+\beta})$ (see Figure 1). Over G', we consider an execution Ξ' such that, agents in G'_1 and G'_2 behave similarly to Ξ until C_t occurs in G'_1 and G'_2, and then make interactions so that Ξ' satisfies global fairness. Since Ξ is globally-fair, we can show the following facts after G'_1 and G'_2 reach C_t in Ξ'.

- v'_α has state s_α as long as $v'_{n+\beta}$ has a state in S_β.
- $v'_{n+\beta}$ has a state in S_β as long as v'_α has state s_α.

From these facts, in Ξ', v'_α continues to have state s_α and $v'_{n+\beta}$ continues to have a state in S_β. Hence, in Ξ', each agent in V'_1 cannot notice the existence of agents in V'_2, and vice versa. This implies that, in stable configurations, $\#red(V) = \#red(V'_1) = \#red(V'_2)$ and $\#blue(V) = \#blue(V'_1) = \#blue(V'_2)$ hold. Since the number of agents in G is odd, $\#red(V) - \#blue(V) = 1$ or $\#blue(V) - \#red(V) = 1$ holds in stable configurations of Ξ. Thus, in stable configurations of Ξ', $|\#red(V') - \#blue(V')| = 2$ holds. Since Ξ' is globally-fair, this is a contradiction. ◀

Now we prove impossibility of an asymmetric protocol with three states. The outline of the proof is as follows. For the purpose of contradiction, we assume that there exists a protocol *Alg* that solves the problem with three states. From Lemma 6, in any globally-fair execution, some agents change their state infinitely often. Now, with three states, the number of *red* or *blue* states is at least one and thus, if we assume without loss of generality that the number of *blue* states is one, agents with the *blue* state change their color eventually after a stable configuration. This is a contradiction.

▶ **Theorem 7.** *There exists no uniform bipartition protocol with three states and designated initial states over arbitrary communication graphs assuming global fairness.*

4.3 Lower Bound for Symmetric Protocols under Global Fairness

In this section, we show that, with arbitrary communication graphs, designated initial states, and no base station assuming global fairness, there exists no symmetric protocol with four states. Recall that, with designated initial states and no base station, clearly any symmetric protocol never solves the problem if the number of agents n is two. Thus, we assume that $3 \leq n \leq P$ holds, where P is a known upper bound of the number of agents. Note that the symmetric protocol proposed in subsection 4.1 solves the problem for $3 \leq n \leq P$.

▶ **Theorem 8.** *There exists no symmetric protocol for the uniform bipartition with four states and designated initial states over arbitrary graph assuming global fairness when P is twelve or more.*

For the purpose of contradiction, suppose that there exists such a protocol Alg. Let R (resp., B) be a state set such that, for any $s \in R$ (resp., $s' \in B$), $f(s) = red$ (resp., $f(s') = blue$) holds. First, we show that the following lemma holds from Lemma 6.

▶ **Lemma 9.** *$|R| = |B|$ holds (i.e., $|R| = 2$ and $|B| = 2$ hold).*

Let ini_r and r (resp., ini_b and b) be states belonging to R (resp., B). In addition, without loss of generality, assume that ini_r is the initial state of agents. Then, we can prove the following lemma.

▶ **Lemma 10.** *There exists some $s_b \in B$ such that $(ini_r, ini_r) \to (s_b, s_b)$ and $(s_b, s_b) \to (ini_r, ini_r)$ hold.*

Without loss of generality, assume that $(ini_r, ini_r) \to (ini_b, ini_b)$ and $(ini_b, ini_b) \to (ini_r, ini_r)$ exist. For some population V, we denote the number of agents with ini_r (resp., ini_b) belonging to V as $\#ini_r(V)$ (resp., $\#ini_b(V)$). Moreover, let $\#ini(V)$ be the sum of $\#ini_r(V)$ and $\#ini_b(V)$. When V is clear from the context, we simply denote them as $\#ini_r$, $\#ini_b$, and $\#ini$, respectively. Then, we can prove the following lemmas and corollary.

▶ **Lemma 11.** *There does not exist a transition rule such that $\#ini$ increases after the transition.*

▶ **Lemma 12.** *Consider a globally-fair execution Ξ of Alg with some complete communication graph G. After some configuration in Ξ, $\#ini \leq 1$ holds.*

▶ **Corollary 13.** *Consider a state set $Ini = \{ini_r,\ ini_b\}$. When $s_1 \notin Ini$ or $s_2 \notin Ini$ holds, if transition rule $(s_1, s_2) \to (s'_1, s'_2)$ exists then $f(s_1) = f(s'_1)$ and $f(s_2) = f(s'_2)$ hold.*

From now on, we prove Theorem 8. Consider a globally-fair execution $\Xi = C_0, C_1, C_2, \ldots$ of Alg with a ring communication graph $G = (V, E)$ such that the number of agents is three, where $V = \{v_0, v_1, v_2\}$. In a stable configuration of Ξ, either $\#blue(V) - \#red(V) = 1$ or $\#red(V) - \#blue(V) = 1$ holds.

First, consider the case of $\#blue(V) - \#red(V) = 1$.

By Lemma 6, red agents keep exchanging r for ini_r in Ξ. Moreover, by Lemma 12, there exists a stable configuration in Ξ such that $\#ini \leq 1$ holds. From these facts, there exists a stable configuration C_t of Ξ such that there exists exactly one agent that has ini_r. Without loss of generality, we assume that the agent is v_0.

Consider the communication graph $G' = (V', E')$ that includes four copies of G. The details of G' are as follows:

- Let $V' = \{v'_0,\ v'_1,\ v'_2,\ v'_3,\ \ldots,\ v'_{11}\}$. Moreover, we define a partition of V' as $V'_1 = \{v'_0,\ v'_1,\ v'_2\}$, $V'_2 = \{v'_3,\ v'_4,\ v'_5\}$, $V'_3 = \{v'_6,\ v'_7,\ v'_8\}$, and $V'_4 = \{v'_9,\ v'_{10},\ v'_{11}\}$. Additionally, let $V'_{red} = \{v'_0,\ v'_3,\ v'_6,\ v'_9\}$ be a set of agents that will have state ini_r.
- $E' = \{(v'_x, v'_y),\ (v'_{x+3}, v'_{y+3}),\ (v'_{x+6}, v'_{y+6}),\ (v'_{x+9}, v'_{y+9}) \in V' \times V' \mid (v_x, v_y) \in E\} \cup \{(v'_x, v'_y) \in V' \times V' \mid x,\ y \in \{0,\ 3,\ 6,\ 9\}\}$.

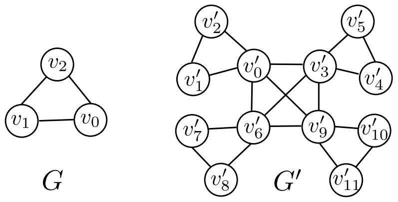

Figure 2 An image of graphs G and G'.

An image of G and G' is shown in Figure 2.

Consider the following execution $\Xi' = C'_0, C'_1, C'_2, \ldots$ of *Alg* with $G' = (V', E')$.

- For $i \le t$, when v_x interacts with v_y at $C_i \to C_{i+1}$, v'_x interacts with v'_y at $C'_{4i} \to C'_{4i+1}$, v'_{x+3} interacts with v'_{y+3} at $C'_{4i+1} \to C'_{4i+2}$, v'_{x+6} interacts with v'_{y+6} at $C'_{4i+2} \to C'_{4i+3}$, and v'_{x+9} interacts with v'_{y+9} at $C'_{4i+3} \to C'_{4i+4}$.
- After C'_{4t}, make interactions between agents in V'_{red} until agents in V'_{red} converge and $\#ini(V'_{red}) \le 1$ holds. We call the configuration $C'_{t'}$.
- After $C'_{t'}$, make interactions so that Ξ' satisfies global fairness.

Until C'_{4t}, agents in V'_1, V'_2, V'_3, and V'_4 behave similarly to agents in V from C_0 to C_t. This implies that, in C'_{4t}, every agent in V'_{red} has state ini_r. From Lemma 12, since ini_r is the initial state of agents, it is possible to make interactions between agents in V'_{red} until agents in V'_{red} converge and $\#ini(V'_{red}) \le 1$ holds. Moreover, since v_0 is the only agent that has ini_r in C_t, no agent in $V'_i \backslash V'_{red} (1 \le i \le 4)$ has state ini_r or ini_b in C'_{4t}. Hence, $\#ini \le 1$ holds in $C'_{t'}$. By Corollary 13, if $\#ini \ge 2$ does not hold, no agent can change its color. Thus, since $\#ini \le 1$ holds after $C'_{t'}$ by Lemma 11, no agent can change its color after $C'_{t'}$. Since v_1 and v_2 are *blue* in C_t, v'_1, v'_2, v'_4, v'_5, v'_7, v'_8, v'_{10}, and v'_{11} are *blue* in $C'_{t'}$. In addition, $\#blue(V'_{red}) = \#red(V'_{red})$ holds. Hence, $\#blue(V') - \#red(V') = 8$ holds. Since no agent can change its color after $C'_{t'}$ and Ξ' is globally-fair, this is a contradiction.

Next, consider the case of $\#red(V) - \#blue(V) = 1$. In this case, we can prove in the same way as the case of $\#blue(V) - \#red(V) = 1$. However, in the case, we focus on ini_b instead of ini_r. That is, we assume that agents in V'_{red} (i.e., v'_0, v'_3, v'_6, and v'_9) have ini_b in C'_{4t}. From C'_{4t}, we make v'_0 (resp., v'_6) interact with v'_3 (resp., v'_9) once. Then, by Lemma 10, all of them transition to ini_r. After that, since all agents in V'_{red} have ini_r, we can construct an execution such that only agents in V'_{red} interact and eventually $\#ini(V'_{red}) \le 1$ holds. As a result, we can lead to contradiction in the same way as the case of $\#blue(V) - \#red(V) = 1$.

4.4 Impossibility under Weak Fairness

In this subsection, assuming arbitrary communication graphs and designated initial states and no base station, we show that there is no protocol that solves the problem under weak fairness. Fischer and Jiang [21] proved the impossibility of leader election for a ring communication graph. We borrow their proof technique and apply it to the impossibility proof of a uniform bipartition problem.

The sketch of the proof is as follows: For the purpose of contradiction, let us assume that there exists such a protocol *Alg*. Consider an execution Ξ of *Alg* for a ring R_1 with three agents v_0, v_1, and v_2. Without loss of generality, we assume that $\#red = 1$ and $\#blue = 2$ hold in a stable configuration of Ξ. After that, consider an execution Ξ' of *Alg* for a ring R_2 with six agents v'_0, v'_1, v'_2, v'_3, v'_4, and v'_5 (see Figure 3). We construct Ξ' such that each agent behaves similarly to Ξ. Concretely, v'_i and v'_{i+3} $(0 \le i \le 3)$ behave similarly to v_i. If

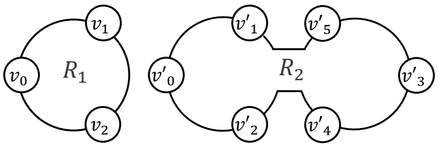

Figure 3 Ring graphs R_1 and R_2.

v_0 interacts with v_1 (resp., v_2) in Ξ, v_0' interacts with v_1' (resp., v_2') and v_3' interacts with v_4' (resp., v_5') in Ξ'. Similarly, If v_1 (resp., v_2) interacts with v_0 in Ξ, v_1' (resp., v_2') interacts with v_0' and v_4' (resp., v_5') interacts with v_3' in Ξ'. If v_1 interacts with v_2 in Ξ, v_1' interacts with v_5' and v_4' interacts with v_2' in Ξ'. Similarly, if v_2 interacts with v_1 in Ξ, v_5' interacts with v_1' and v_2' interacts with v_4' in Ξ'. Observe that, if $s(v_i) = s(v_i') = s(v_{i+3}')$ holds before the interactions for $0 \le i \le 2$, $s(v_i) = s(v_i') = s(v_{i+3}')$ holds even after the interactions. Thus, since $s(v_i) = s(v_i') = s(v_{i+3}')$ holds in the initial configuration, $s(v_i) = s(v_i') = s(v_{i+3}')$ continues to hold. Hence, in the stable configuration of Ξ', $\#red = 2$ and $\#blue = 4$ hold. This contradicts that *Alg* solves the problem. Therefore, we have the following theorem.

▶ **Theorem 14.** *There exists no protocol that solves the uniform bipartition problem with designated initial states and no base station under weak fairness assuming arbitrary communication graphs.*

5 Concluding Remarks

In this paper, we consider the uniform bipartition problem with designated initial states assuming arbitrary communication graphs. We investigated the problem solvability, and even provided tight bounds (with respect to the number of states per agent) in the case of global fairness.

Our work raises interesting open problems:

- Is there a relation between the uniform bipartition problem and other classical problems such as counting, leader election, and majority? We pointed out the reuse of some proof arguments, but the existence of a more systematic approach is intriguing.
- What is the time complexity of the uniform bipartition problem?

References

1 Dan Alistarh, James Aspnes, David Eisenstat, Rati Gelashvili, and Ronald L Rivest. Time-space trade-offs in population protocols. In *Proc. of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2560–2579, 2017.

2 Dan Alistarh, James Aspnes, and Rati Gelashvili. Space-optimal majority in population protocols. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2221–2239. SIAM, 2018.

3 Dan Alistarh and Rati Gelashvili. Polylogarithmic-time leader election in population protocols. In *Proc. of the 42nd International Colloquium on Automata, Languages, and Programming*, pages 479–491, 2015.

4 Dana Angluin, James Aspnes, Melody Chan, Michael J Fischer, Hong Jiang, and René Peralta. Stably computable properties of network graphs. In *Proc. of International Conference on Distributed Computing in Sensor Systems*, pages 63–74, 2005.

5 Dana Angluin, James Aspnes, Zoë Diamadi, Michael J Fischer, and René Peralta. Computation in networks of passively mobile finite-state sensors. *Distributed computing*, 18(4):235–253, 2006.

6 Dana Angluin, James Aspnes, and David Eisenstat. A simple population protocol for fast robust approximate majority. *Distributed Computing*, 21(2):87–102, 2008.

7 Dana Angluin, James Aspnes, Michael J Fischer, and Hong Jiang. Self-stabilizing population protocols. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 3(4):13, 2008.

8 James Aspnes, Joffroy Beauquier, Janna Burman, and Devan Sohier. Time and space optimal counting in population protocols. In *Proc. of International Conference on Principles of Distributed Systems*, pages 13:1–13:17, 2016.

9 Joffroy Beauquier, Peva Blanchard, and Janna Burman. Self-stabilizing leader election in population protocols over arbitrary communication graphs. In *International Conference on Principles of Distributed Systems*, pages 38–52. Springer, 2013.

10 Joffroy Beauquier, Janna Burman, Simon Claviere, and Devan Sohier. Space-optimal counting in population protocols. In *Proc. of International Symposium on Distributed Computing*, pages 631–646, 2015.

11 Joffroy Beauquier, Julien Clement, Stephane Messika, Laurent Rosaz, and Brigitte Rozoy. Self-stabilizing counting in mobile sensor networks with a base station. In *Proc. of International Symposium on Distributed Computing*, pages 63–76, 2007.

12 Stav Ben-Nun, Tsvi Kopelowitz, Matan Kraus, and Ely Porat. An o (log3/2 n) parallel time population protocol for majority with o (log n) states. In *Proceedings of the 39th Symposium on Principles of Distributed Computing*, pages 191–199, 2020.

13 Petra Berenbrink, Robert Elsässer, Tom Friedetzky, Dominik Kaaser, Peter Kling, and Tomasz Radzik. A population protocol for exact majority with o(log5/3 n) stabilization time and theta(log n) states. In *32nd International Symposium on Distributed Computing, DISC 2018, New Orleans, LA, USA, October 15-19, 2018*, volume 121 of *LIPIcs*, pages 10:1–10:18, 2018.

14 Petra Berenbrink, Robert Elsässer, Tom Friedetzky, Dominik Kaaser, Peter Kling, and Tomasz Radzik. Time-space trade-offs in population protocols for the majority problem. *Distributed Computing*, pages 1–21, 2020.

15 Petra Berenbrink, George Giakkoupis, and Peter Kling. Optimal time and space leader election in population protocols. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pages 119–129, 2020.

16 Petra Berenbrink, Dominik Kaaser, and Tomasz Radzik. On counting the population size. In *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing*, pages 43–52, 2019.

17 Olivier Bournez, Jérémie Chalopin, Johanne Cohen, Xavier Koegler, and Mikael Rabie. Population protocols that correspond to symmetric games. *International Journal of Unconventional Computing*, 9, 2013.

18 Davide Canepa and Maria Gradinariu Potop-Butucaru. Self-stabilizing tiny interaction protocols. In *Proceedings of the Third International Workshop on Reliability, Availability, and Security*, page 10. ACM, 2010.

19 Carole Delporte-Gallet, Hugues Fauconnier, Rachid Guerraoui, and Eric Ruppert. When birds die: Making population protocols fault-tolerant. *Distributed Computing in Sensor Systems*, pages 51–66, 2006.

20 David Doty and David Soloveichik. Stable leader election in population protocols requires linear time. *Distributed Computing*, 31(4):257–271, 2018.

21 Michael Fischer and Hong Jiang. Self-stabilizing leader election in networks of finite-state anonymous agents. In *International Conference On Principles Of Distributed Systems*, pages 395–409. Springer, 2006.

22 Leszek Gąsieniec, David Hamilton, Russell Martin, Paul G Spirakis, and Grzegorz Stachowiak. Deterministic population protocols for exact majority and plurality. In *Proc. of International Conference on Principles of Distributed Systems*, pages 14:1–14:14, 2016.

23 Leszek Gąsieniec, Grzegorz Stachowiak, and Przemyslaw Uznanski. Almost logarithmic-time space optimal leader election in population protocols. In *The 31st ACM on Symposium on Parallelism in Algorithms and Architectures*, pages 93–102. ACM, 2019.

24 Tomoko Izumi, Keigo Kinpara, Taisuke Izumi, and Koichi Wada. Space-efficient self-stabilizing counting population protocols on mobile sensor networks. *Theoretical Computer Science*, 552:99–108, 2014.

25 Adrian Kosowski and Przemyslaw Uznanski. Brief announcement: Population protocols are fast. In *Proceedings of the 2018 ACM Symposium on Principles of Distributed Computing*, pages 475–477, 2018.

26 Anissa Lamani and Masafumi Yamashita. Realization of periodic functions by self-stabilizing population protocols with synchronous handshakes. In *Proc. of International Conference on Theory and Practice of Natural Computing*, pages 21–33, 2016.

27 George B Mertzios, Sotiris E Nikoletseas, Christoforos L Raptopoulos, and Paul G Spirakis. Determining majority in networks with local interactions and very small local memory. In *International Colloquium on Automata, Languages, and Programming*, pages 871–882. Springer, 2014.

28 Satoshi Murata, Akihiko Konagaya, Satoshi Kobayashi, Hirohide Saito, and Masami Hagiya. Molecular robotics: A new paradigm for artifacts. *New Generation Computing*, 31(1):27–45, 2013.

29 Yuichi Sudo, Fukuhito Ooshita, Taisuke Izumi, Hirotsugu Kakugawa, and Toshimitsu Masuzawa. Time-optimal leader election in population protocols. *IEEE Transactions on Parallel and Distributed Systems*, 2020.

30 Yuichi Sudo, Fukuhito Ooshita, Hirotsugu Kakugawa, and Toshimitsu Masuzawa. Loosely-stabilizing leader election on arbitrary graphs in population protocols. In *International Conference on Principles of Distributed Systems*, pages 339–354. Springer, 2014.

31 Tomoki Umino, Naoki Kitamura, and Taisuke Izumi. Differentiation in population protocols. *6th workshop on biological distributed algorithms(BDA)*, 2018.

32 Hiroto Yasumi, Naoki Kitamura, Fukuhito Ooshita, Taisuke Izumi, and Michiko Inoue. A population protocol for uniform k-partition under global fairness. *International Journal of Networking and Computing*, 9(1):97–110, 2019.

33 Hiroto Yasumi, Fukuhito Ooshita, and Michiko Inoue. Uniform partition in population protocol model under weak fairness. *the 23rd International Conference on Principles of Distributed Systems*, 2019.

34 Hiroto Yasumi, Fukuhito Ooshita, Michiko Inoue, and Sébastien Tixeuil. Uniform bipartition in the population protocol model with arbitrary communication graphs, 2020. `arXiv:2011.08366`.

35 Hiroto Yasumi, Fukuhito Ooshita, Ken'ichi Yamaguchi, and Michiko Inoue. Constant-space population protocols for uniform bipartition. *the 21st International Conference on Principles of Distributed Systems*, 2017.

36 Hiroto Yasumi, Fukuhito Ooshita, Ken'ichi Yamaguchi, and Michiko Inoue. Space-optimal population protocols for uniform bipartition under global fairness. *IEICE TRANSACTIONS on Information and Systems*, 102(3):454–463, 2019.