Vol.2010-AL-132 No.4
2010/11/19

自己安定リーダー選挙MPPにおける領域複雑度の上下界について

溝口 隆[†1] 小野廣隆[†1]
来嶋秀治[†1] 山下雅史[†1]

本論文は，自己安定リーダー選挙メディエイテッドポピュレーションプロトコル (SS-LE MPP) の領域複雑度に関する研究結果である．Cai, Izumi and Wada (2009) は n エージェントに対する自己安定リーダー選挙ポピュレーションプロトコル (SS-LE PP) は少なくとも n 個のエージェント状態を必要とすることを示した．さらに n エージェントに対する SS-LE MPP のうちエージェント状態数が n であるプロトコルを与えた．MPP は Chatzigiannakis, Michail and Spirakis (2009) によって紹介された．MPP は分散ネットワークモデルの 1 つで，各エージェント間に局所的なメモリを追加した PP の拡張モデルである．一般的に MPP は PP よりも計算能力が高いことが知られている．一方で SS-LE のエージェント状態に対する空間複雑度の減少可能性については知られていない．本稿では，n エージェントの SS-LE MPP のうちエージェント状態数が $n-1$ であり，各エージェント間に 1 ビットのメモリが与えられたプロトコルを与える．

On Space Complexity of Self-Stabilizing Leader Election in Mediated Population Protocol

Ryu Mizoguchi,[†1] Hirotaka Ono,[†1] Shuji Kijima[†1] and Masafumi Yamashita [†1]

This paper investigates the space complexity of a self stabilizing leader election in a mediated population protocol (SS-LE MPP). Cai, Izumi and Wada (2009) showed that SS-LE in a population protocol (SS-LE PP) for n agents requires at least n agent-states, and gave a SS-LE PP with n agent-states for n agents. MPP is a model of distributed computation, introduced by Chatzigiannakis, Michail and Spirakis (2009) as an extension of PP allowing an extra memory on every agents pair. While they showed that MPP is stronger than PP in general, it was not known if a MPP can really reduce the space complexity of SS-LE with respect to agent-states. We in this paper give a SS-LE MPP with $n-1$ agent-states and a single bit memory on every agents pair for n agents.

1. Introduction

Population Protcol (*PP*), proposed by Angluin et al.[1], is a model of distributed computation consisting of agents and communication links among them, and *Mediated Population Protocol* (*MPP*) proposed by Chatzigiannakis et al.[7], is an extended model of PP allowing memories on communication links. PP and MPP are models of sensor networks consisting of *passively* mobile agents with limited computational resources, motivated by practical networks such as networks of smart sensors attached to cars or animals, synthesis of chemical materials, complex biosystems, and so on (cf.[1),7)]).

In MPP, every agent is identically programmed as a finite state machine, and every communication link is equipped with a (finite) buffer. The agents sequentially interact with each other updating their states; a pair of agents chosen by a scheduler updates their own *agent-states* and *edge-states* between them in an interaction. The order of interactions of agent-pairs is unpredictable, and is scheduled by an adversarial scheduler satisfying a *fairness condition*; the scheduler must accept any possible interaction within a finite time if a configuration in which the interaction can arise should appear infinitely many times.

Angluin et al.[3] discussed the *leader election* in a population protocol, which is a fundamental problem in distributed computing, and introduced the problem of *self stabilizing leader election* in a *population protocol* (*SS-LE PP*, for short). In a SS-LE PP, any initial configuration of agent-states eventually have to reach at a configuration whose successive configurations contain exactly one leader. Thus a SS-LE PP should be equipped with seemingly conflicting functions; the protocol has to decrease the number of leaders if a configuration contains two or more leaders, while the protocol has to appoint an agent to be a leader if a configuration does not contain a leader. This causes

†1 九州大学
Kyushu University

some difficulties on SS-LE, as it is usual with self-stabilizing distributed problems.

Angluin et al.[2] discussed that no SS-LE PP with any constant number of agent-states exists for general n agents on some types of interaction graphs. Fischer and Jiang[9] discussed SS-LE PP assuming a (global) oracle for leader detector on complete communication graphs. Canepa and Potop-Butucaru[6] discussed SS-LE PP on any communication graph in the same assumption with[9]. Cai, Izumi and Wada[5] discussed SS-LE on complete interaction graph without any (global) oracles, and showed that SS-LE for n agents requires at least n agent-states, and presented a SS-LE PP with n agent-states for n agents.

This paper is concerned with the space complexity of *self stabilizing leader election* in a *mediated population protocol* (*SS-LE MPP*, for short) for n agents, where we assume that an interaction graph is complete as did Cai, Izumi and Wada[5]. We present a SS-LE MPP with $n-1$ agent-states and two edge-states for n agents. As far as the authors know, this is the first result on SS-LE MPP. One may say it obvious that the number of agent-stats decreases in MPP comparing with PP due to extra memories on edges. In fact, it is clear that n is also sufficient for the number of agent-states in SS-LE MPP for n agents. However, extra memories on edges in MPP, which are expected to resolve the issue of conflicting functions in the self-stabilizing setting instead of a certain number of agent-states, may cause another issue of increasing possible (bad) initial configurations in the self-stabilizing setting.

This paper is organized as follows; in Section 2, we describe the detail of our model. We in Section 3 give a SS-LE MPP with $n-1$ agent-states and 2 edge-states for n agents.

2. Model Description — SS-LE MPP

A *mediated population protocol* is defined by 3-tuple (Q, S, δ), where Q denotes a finite set of agent-states, S denotes a finite set of edge states, and $\delta \colon Q \times Q \times S \to Q \times Q \times S$ denotes a transition function. Let A denote the set of anonymous agents and let $n = |A|$, and let $\mathcal{C} \stackrel{\text{def}}{=} Q^A \times S^{\binom{A}{2}}$ denote *all configurations*. A transition from a configuration $C \in \mathcal{C}$ to the next configuration $C' \in \mathcal{C}$ is defined as follows. An arbitrary pair of agents $a_i, a_j \in A (a_i \neq a_j)$ is chosen by a scheduler, thus an interaction graph is complete in our model. States of the agents a_i and a_j, and a state of an edge $\{a_i, a_j\}$ are updated according to a transition function δ. Let $r : (p, q, s) \mapsto (p', q', s')$ denote a specified transition rule of δ, and let $C \xrightarrow{r; a_i, a_j} C'$ denote a transition from $C \in \mathcal{C}$ to $C' \in \mathcal{C}$ in which agents a_i and a_j interact and their states p, q and edge-state s between them are updated to p', q', s' according to the rule r of δ. We simply write $C \xrightarrow{r} C'$ without confusing. An *execution* of a protocol is represented by an infinite sequence of configurations and transitions $C_0, r_0, C_1, r_1, \ldots$, where C_0 is an initial configuration and $C_i \xrightarrow{r_i} C_{i+1} (i \geq 0)$.

We assume that a scheduler in a MPP is *adversarial* but (globally) *fair*, as usual (cf.[5]). Thus we have to think that an adversarial scheduler schedules the order of interactions in a worst case scenario for us, but it is forced to satisfy that if a configuration $C \in \mathcal{C}$ appears infinitely often in an execution, a configuration $C' \in \mathcal{C}$ must also appear infinitely often in an execution, where C' is a configuration obtained by an arbitrary transition $r \in \delta$ which arises in C. We say that C *eventually transits* to C', denoted by $C \xrightarrow{*} C'$, if C' must appear after C by the adversarial but globally fair scheduler in MPP. In addition, we describe a sequence of transitions as the *trace* T.

Leader election in a MPP is to assign a special state, representing a "leader", in S to exactly one agent. We say a configuration $C \in \mathcal{C}$ is *legal* if C contains exactly one agent with the leader state, and so does any configuration C' satisfying $C \xrightarrow{*} C'$. Let $\mathcal{L}$ denote the set of all legal configurations. We say a protocol for the leader election (for a distributed problem, in general), is *self-stabilizing* if $C \xrightarrow{*} C', C' \in \mathcal{L}$ hold for any $C \in \mathcal{C}$. We simply say *SS-LE MPP* as a mediated population protocol for the leader election which is self stabilizing.

Our goal is to give upper bound of the sizes of the agent-states Q and edge-states S for SS-LE MPP concerning the number of agents n. Main results of the paper are to give a SS-LE MPP with $|Q| = n-1$ and $|S| = 2$ for n agents in Section 3.

3. Simple SS-LE MPP with $n-1$ Agent-states

In this section, we show the following.

Theorem 1. *There exists a SS-LE MPP with $n-1$ agent-states and 2 edge-states for $n(\geq 4)$ agents.*

We give a constructive proof. In particular, we show that Protocol P_1, defined as follows, is a SS-LE MPP.

Protocol P_1

$Q = \{q_0, q_1, \ldots, q_{n-2}\}$, where q_0 denotes the leader state.

$S = \{s_0, s_1\}$,

$\delta = \{$

$r_1: (q_0, q_0, s) \mapsto (q_0, q_{n-2}, s_0)$ for $s \in S$,

$r_2: (q_1, q_1, s) \mapsto (q_1, q_2, s_0)$ for $s \in S$,

$r_3: (q_2, q_2, s_0) \mapsto (q_2, q_2, s_1)$,

$r_4: (q_2, q_2, s_1) \mapsto (q_2, q_1, s_0)$,

$r_5: (q_2, q_1, s_1) \mapsto (q_2, q_0, s_0)$ $((q_1, q_2, s_1) \mapsto (q_0, q_2, s_0)$, symmetrically$)$,

$r_6: (q_i, q_i, s) \mapsto (q_i, q_{i-1}, s_0)$ for $i \geq 3, s \in S$,

$r_7: (q_j, q_k, s) \mapsto (q_j, q_k, s_0)$ for $j \neq k, s \in S$, except for the case of r_5

$\}$.

Remark. *Except for Transition r_5, the state of an agent can change only when the agent interacts with another agent in the same state.*

Let $\gamma_k(C)$ for $k \in \{0, 1, \ldots, n-2\}$ denote the number of agents with state q_k in a configuration $C \in \mathcal{C}$. We define a set of configurations $\mathcal{L} \subset \mathcal{C}$ by

$$\mathcal{L} \stackrel{\text{def}}{=} \left\{ C \in \mathcal{C} \;\middle|\; \begin{array}{l} \gamma_k(C) > 0 \text{ for } k \in \{0,1,\ldots,n-2\},\ \gamma_1(C) + \gamma_2(C) = 3, \\ \text{both ends of an edge with state } s_1 \text{ are agents with state } q_2. \end{array} \right\}.$$

Note that the number of edges with state s_1 in $C \in \mathcal{L}$ is at most one since $\gamma_2(C)$ is at most two from the definition of $\mathcal{L}$.

In the following, we claim that $\mathcal{L}$ is the set of legal configurations for Protocol P_1. Let H denote a subconfiguration of $C \in \mathcal{L}$ consisting of three agents with states q_1 or q_2 and three edges among them. Then H can be one of three types of subconfigurations H_1, H_2, H_3 of six possible types $H_1, H_2, H_3, H_4, H_5, H_6$ in Fig.1 which satisfy that the number of edges with state s_1 is at most one, $\gamma_1(C) > 0$ and $\gamma_2(C) > 0$. First, we show that $\mathcal{L}$ is "closed" under the transition function δ.

Lemma 2. *If configurations C and C' satisfy $C \in \mathcal{L}$ and $C \xrightarrow{*} C'$, then $C' \in \mathcal{L}$ and $C' \xrightarrow{*} C$ hold.*

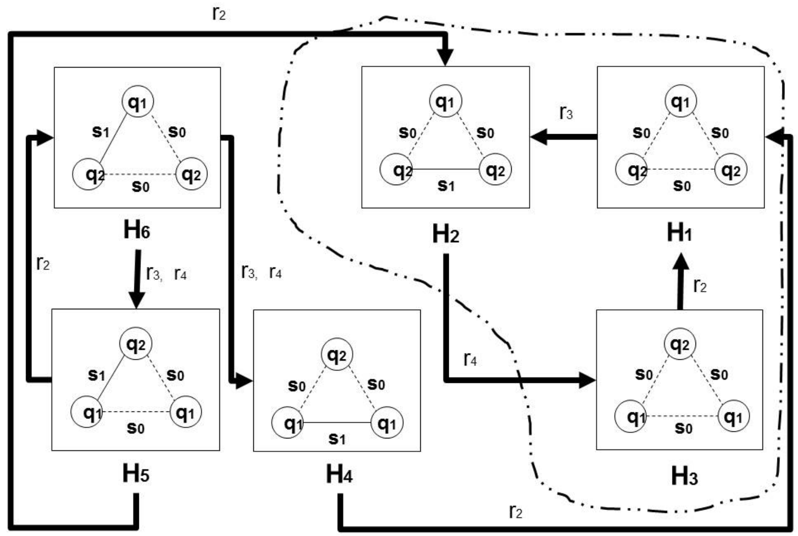

図 1 $\{H_1, H_2, H_3\}$ is closed.

Proof. Transition r_1 cannot arise in a configuration $C \in \mathcal{L}$ by the condition $\gamma_0(C) = 1$ in $\mathcal{L}$. By the condition $\gamma_1(C) + \gamma_2(C) = 3$ and $\gamma_k(C) > 0$ for $k \in \{0, 1, \ldots, n-2\}$, C satisfies $\gamma_k(C) = 1$ for $k \in \{3, 4, \ldots, n-2\}$. Thus, Transition r_6 cannot arise in $C \in \mathcal{L}$.

Now we show that Transition r_5 cannot arise in C. Since the edge-state s_1 appears only within the subconfiguration H of $C \in \mathcal{L}$, it is enough to show that Transition r_5 cannot arise in subconfigurations H_1, H_2, H_3 of C.

Case 1. $H = H_1$: Consider an agent with state q_1 as a and agents with state q_2 as b, c, and consider every state of every edge among them as state s_0. Then Transitions r_3 or r_7 can arise in H_1, that is $H_1 \xrightarrow{r_3} H_2$ or $H_1 \xrightarrow{r_7} H_1$.

Case 2. $H = H_2$: Consider an agent with state q_1 as a and agents with state q_2 as b, c, and consider a state of the only one edge between the two agents b, c as state s_1. Then

Transitions r_4 or r_7 can arise in H_2, that is $H_2 \xrightarrow{r_4} H_3$ or $H_2 \xrightarrow{r_7} H_2$.

Case 3. $H = H_3$: Consider agents with state q_1 as a, b and an agent with state q_2 as c and consider every state of every edge among them as state s_0. Then Transitions r_2 or r_7 can arise in H_3, that is $H_3 \xrightarrow{r_2} H_1$ or $H_3 \xrightarrow{r_7} H_3$.

Therefore, if $C \in \mathcal{L}$ and C' satisfies $C \xrightarrow{*} C'$, then $C' \in \mathcal{L}$ and $C' \xrightarrow{*} C$ hold. □

Next, we show that any configuration $C \in \mathcal{C}$ eventually transits to a configuration $C' \in \mathcal{L}$, in Lemma 5. To show Lemma 5, we show Lemmas 3 and 4.

Lemma 3. *If a configuration $C \in \mathcal{C}$ satisfies $\gamma_k(C) > 0$ for $k \in \{0, 2, 3, \ldots, n-2\}$, and $C \xrightarrow{*} C'$, then the configuration C' also satisfies $\gamma_k(C') > 0$.*

Proof. After an agent with state q_k for $k \in \{0, 2, 3, \ldots, n-2\}$ interacts with any other agent, $\gamma_k(C)$ decreases at most one in any transition. In fact, $\gamma_k(C)$ decreases only when the agent interacts with another agent in the same state q_k. This implies that $\gamma_k(C)$ never decreases from one to zero by any transition. □

Lemma 4. *If configurations $C, C' \in \mathcal{C}$ satisfy $\gamma_0(C) = 0$, $C \xrightarrow{*} C'$ and $\gamma_0(C') = 0$, then the followings hold;*
(1) $\sum_{i=1}^{k} \gamma_i(C') \geq \sum_{i=1}^{k} \gamma_i(C)$ *for any* $k \in \{2, 3, \ldots, n-2\}$.
(2) *If* $\gamma_i(C) > 0$, *then* $\gamma_i(C') > 0$.

Proof. 1. If $\gamma_0(C) = 0$ and $\gamma_0(C') = 0$ hold, Transition r_5 cannot have arisen on $C \xrightarrow{*} C'$. Note that Transition r_7 does not change any agent-state. Since $n-2$ states are assigned to n agents, there exists a pair of agents and they are in a common state q_i. When $i \geq 3$, Transitions r_6 or r_7 can arise in C and exactly one of the agents changes its state from q_i to q_{i-1}. Thus $\gamma_{i-1}(C') + \gamma_i(C') = \gamma_{i-1}(C) + \gamma_i(C)$ and $\gamma_{i-1}(C') = \gamma_{i-1}(C) + 1$. When $i = 1, 2$, Transitions r_2, r_3 or r_4 can arise in C except for Transitions r_5 and r_7 and their transitions does not change $q \in \{q_1, q_2\}$ to $q' \notin \{q_1, q_2\}$. Therefore, for any $r \in \{r_2, r_3, r_4\}$ a configuration C' of $C \xrightarrow{r} C'$ satisfies that $\gamma_1(C) + \gamma_2(C) = \gamma_1(C') + \gamma_2(C')$. That indicates $\sum_{i=1}^{k} \gamma_i(C') \geq \sum_{i=1}^{k} \gamma_i(C)$.

2. By Lemma 3, if $\gamma_0(C) = 0$ and $\gamma_0(C') = 0$, Transition r_5 cannot have arisen on $C \xrightarrow{*} C'$. In arbitrary transitions except for r_5, $\gamma_1(C)$ decreases at most one in a transition. $\gamma_1(C)$ decreases only when a pair of agents with same state q_1 interact. This implies that $\gamma_1(C)$ never decreases from one to zero by any transition. □

Lemma 5. *For any configuration $C \in \mathcal{C}$, there exists a configuration $C' \in \mathcal{L}$ and $C \xrightarrow{*} C'$.*

Proof. Case 1. $\gamma_0(C) = 0$

We show that for any configuration $C \in \mathcal{C}$, there exists a configuration $C' \in \mathcal{C}$ satisfying that $C \xrightarrow{*} C'$ and $\gamma_0(C') > 0$.

Case 1.1. $\gamma_1(C) + \gamma_2(C) \leq 3$

Since $n-4$ states $q_3, q_4, \ldots, q_{n-2}$ are assigned to at least $n-3$ agents, there exists a pair of agents and their states are common q_i. When $i \geq 3$, Transitions r_6 or r_7 can arise in C and exactly one of the agents changes its state from q_i to q_{i-1}. By Lemma 4, $\gamma_i(C)(> 0)$ does not become zero by any transition and $\sum_{i=1}^{k} \gamma_i(C)$ does not decrease by any transition, thus there exists a configuration $C' \in \mathcal{C}$ satisfying that $C \xrightarrow{*} C'$ and $\gamma_1(C') + \gamma_2(C') > 3$.

Case 1.2. $\gamma_1(C) + \gamma_2(C) > 3$

Suppose Transition r_5 cannot have arisen on $C \xrightarrow{*} C'$, then Transitions r_2, r_3, r_4, or r_7 can arise in C except for Transition r_5. It implies that C eventually transits to a configuration $C' \in \mathcal{C}$ satisfying $\gamma_2(C') \geq 3$, thus configurations satisfying $\gamma_2(C') \geq 3$ infinitely often appear. Consider three agents a, b, c with state q_2. A trace $(r_3; a, b)$, $(r_3; b, c)$, $(r_4; a, b)$, $(r_5; c, b)$ can infinitely often arise in C'. Therefore, it contradicts the assumption of the global fairness, the configuration eventually transits to Case 2.

Case 2. $\gamma_0(C) > 0$

If $\gamma_0(C) > 1$, Transition r_1 can infinitely often have arisen by fairness condition. Hence C eventually transits to a configuration $C' \in \mathcal{C}$ satisfying $\gamma_0(C') = 1$. If $\sum_{i=0}^{k} \gamma_i(C) \geq k+3$ for $k \geq 3$, in a similar way as Case 1, C eventually transits to a configuration $C'' \in \mathcal{C}$ and $\gamma_0(C'') \geq 2$, and $\sum_{i=0}^{k} \gamma_i(C'')$ decreases again after Transition r_1 arises. Since $n-1$ states are assigned to n agents and $\gamma_j(C)(> 0)$ except for $j = 1$ does not become zero by any transitions, C eventually transits to $C' \in \mathcal{C}$ satisfying $\gamma_2(C') = 2$ and $\gamma_j(C') = 1$ except for $j = 2$. By Lemma 3 such a configuration C' infinitely often appears, therefore Transition r_7 can have arisen until no edge with state s_1 remain C'. It is clear that such a configuration is included in $\mathcal{L}$. □

We obtain Theorem 1 by Lemmas 2 and 5.

4. Conclusion

We gave a SS-LE MPP with $n - 1$ agent-states and two edge-states for n agents. Although we get a result of lower bound now, this paper do not include the result. A future work is to analyze SS-LE MPP with a constant edge-states. Analyses on other interaction graphs may be another future work.

参 考 文 献

1) Angluin, D., Aspnes, J., Diamadi, Z., Fischer, M.J., Peralta, R.: Computation in networks of passively mobile finite-state sensors. Distributed Computing, **18** (2006), 235–253.
2) Angluin, D., Aspnes, J., Fischer, M.J., Jiang, H.: Self-stabilizing population protocols. Lecture Notes in Computer Science, **3974** (2005), 103–117.
3) Angluin, D., Aspnes, J., Eisenstat, D.: Stably computable predicates are semilinear. Lecture Notes in Computer Science, **3560** (2005), 63–74.
4) Aspnes, J., Ruppert, E.: An introduction to population protocols. Bulletin of the EATCS, **93** (2007), 98–117.
5) Cai, S., Izumi, T., Wada, K.: Space complexity of self-stabilizing leader election in passively-mobile anonymous agents. Lecture Notes in Computer Science, **5869** (2010), 113–125.
6) Canepa, D., Potop-Butucaru, M.G.: Stabilizing leader election in population protocols. Unpublished (2007).
http://hal.archives-ouvertes.fr/docs/00/16/66/52/PDF/RR-6269.pdf
7) Chatzigiannakis, I., Michail, O., Spirakis, P.G.: Mediated population protocols. Lecture Notes in Computer Science, **5556** (2009), 363–374.
8) Chatzigiannakis, I., Michail, O., Spirakis, P.G.: Recent advances in population protocols. Lecture Notes in Computer Science, **5734** (2009), 56–76.
9) Fischer, M.J., Jiang, H.: Self-stabilizing leader election in networks of finite-state anonymous agent. Lecture Notes in Computer Science, **4305** (2006), 395–409.