有向リング状の個体群モデルにおける
緩自己安定リーダ選挙アルゴリズム

横田 大輔 *　　首藤 裕一 *　　増澤 利光 *

内容梗概

個体群プロトコルモデルは，ある種のモバイルネットワークを表現する計算モデルであり，近年，盛んに研究が行われている．本稿では，有向リング状の個体群プロトコルモデルにおいて，リーダ選挙問題を解く緩自己安定プロトコルを提案する．ここで，緩自己安定プロトコルとは，任意の初期状況から実行を開始してもやがて所望の性質を満たすようになり，それ以降その性質を十分に長いあいだ維持するプロトコルのことを指す．緩自己安定プロトコルとは異なり，所望の性質を回復した後，その性質を永遠に維持するプロトコルを自己安定プロトコルと呼ぶ．

2005 年に Angluin らは，奇数サイズのリングネットワーク上で動作する自己安定リーダ選挙プロトコルを提案した．2019 年に Chen と Chen は，任意のリングネットワーク上で動作する自己安定リーダ選挙プロトコルを提案した．しかしながら，後者のプロトコルは正しい状況（リーダがひとつであるような状況）に収束するまでに指数期待ステップを要する．本稿で提案する緩自己安定プロトコルは，ネットワークサイズの上界 N が既知であるという仮定のもと，任意のリングネットワーク上で，$O(nN\log n)$ 期待ステップでただひとつのリーダを選択し，$\Omega(Ne^N)$ 期待ステップの間そのただひとつのリーダを保持する．

1　はじめに

個体群プロトコルモデルとは，計算能力が非常に制限された多数のセンサノードによって構成されるモバイルセンサネットワークを表現する計算モデルであり Angluin ら [1] によって提案された．本モデルにおいては，ネットワークを構成するノードを**個体**，ネットワークそのものを**個体群**と呼ぶ．本モデルでは，各ノードは他のノードが十分近くに接近した場合にのみ，その 2 つのノード間で通信することができ，その通信によって互いの状態を遷移させていく．本稿では，分散システムにおける重要かつ基本的な問題である**リーダ選挙問題**に対する，リングネットワーク上の個体群プロトコルを提案する．リーダ選挙問題とは，個体群中のただひとつの個体をリーダ状態に，その他の個体を非リーダ状態にすることを目的とする問題である．具体的には，いずれかのただひとつの個体がリーダであることを示す記号 l を出力し，その他のすべての個体は非リーダであることを示す記号 f を出力することを求める問題である．

分散システムは一般に多数のノードで構成されるために，各ノードにおける故障の発生や性能劣化などにより，機能面・性能面で不安定になりやすいという特徴がある．そのため，分散システムに耐故障性を持たせることは重要である．耐故障性を実現する技法のひとつに**自己安定** [4] という概念がある．自己安定の概念は次のように定義される．

1. **収束性**：任意の初期状況から実行を開始しても，システムはやがて安全状況と呼ばれる状況に到達する．
2. **閉包性**：一度システムが安全状況に到達すると，それ以降，システムは永遠に所望の性質を満たし続ける．

自己安定プロトコルは，どのような一時故障などが発生したとしても，自律的に目的の状況に到達できるという優れた耐故障性を持つ．なぜなら，一時故障が終了した直後の状況を新たな初期状況と考えると，収束性により，再び安全状況に到達するからである．しかしながら，自己安定プロトコルは優れた耐故障性を有する一方で，その要件の厳しさから，問題によってはこれを解く自己安定アルゴリズムが存在しない場合やコスト (時間計算量や空間計算量など) が著しく大きくなる場合がある．

この問題点に対処するため，首藤ら [5] は，閉包性の条件を次のように緩めた**緩自己安定**という新たな概念を導入した．

2. **緩閉包性**：一度システムが安全状況に到達すると，それ以降，システムは極めて長い間所望の性質を満たし続ける．

緩自己安定プロトコルにおいて，安全状況は永遠に保たれるわけではないので，**緩安全状況**と呼ばれる．しかしながら，システムが所望の性質を維持する時間が十分に長いのであれば，実用上は自己安定と変わらない耐故障性を持つとみなすことができる．

* 大阪大学大学院情報科学研究科

Angluin ら [2] は，2005 年に個体群プロトコルモデルにおいて，奇数サイズのリンググラフ上でリーダ選挙問題を解決するリーダ選挙自己安定プロトコルを与えた．2019 年に，Chen[3] らはリンググラフの奇数サイズという制約を除去することに成功した．具体的には，任意のサイズの有向リンググラフ上でリーダ選挙問題を解決する自己安定プロトコルを提案した．ただし，そのプロトコルの収束時間の期待値は，個体数を n としたときに $2^{O(n^2 \log n)}$ であり，極めて大きい．本稿では，任意の有向リンググラフにおいて，個体数 n の上界 N が既知であるという仮定のもと，多項式期待時間で収束する緩自己安定リーダ選挙プロトコルを提案する．提案プロトコルは，リーダを削減する役割を持つ**弾丸**とリーダが弾丸に殺されることを防ぐ役割を持つ**バリア**いう考え方を用いることで，任意の初期状況から実行を開始しても $O(nN \log n)$ 期待ステップで唯一のリーダを選出し，それ以降，$\Omega(Ne^N)$ 期待ステップの間，その唯一のリーダを保持する．

2 諸定義

2.1 個体群プロトコル

個体群プロトコルモデルではネットワークを構成するノードを**個体**，複数の個体で構成されるネットワーク全体を**個体群**と呼ぶ．各個体は有限状態機械であり，常に 1 つの状態を持つ．2 つの個体間で**交流**と呼ばれる通信がときおり発生し，その際，両個体の現在の状態とプロトコルが定める状態遷移規則によって自身の状態を更新する．交流が繰り返し行われ，個体の状態が更新されていくことでプロトコルの実行が進行する．個体群は単純有向グラフ $G = (V, E)$ で表す．$V = \{u_0, u_1, \dots, u_{n-1}\}(n \geq 2)$ は個体群を構成する個体の集合である．$E \subseteq V \times V$ は有向辺の集合である．2 つの個体 u,v は $(u, v) \in E$ の場合にのみ u を**呼びかけ側**，v を**応答側**として交流できる．交流における状態の更新において呼びかけ側の個体と応答側の個体は異なる動作を行うことができるので，本モデルにおいてすべての個体は均一であるが，対称性の破壊は本質的な問題とならない．本稿では有向リンググラフを考える．すなわち，$E = \{(u_i, u_{i+1 \bmod n}) \mid i = 0, 1, \dots, n-1\}$ であるような個体群のみを考える．

プロトコル $P(Q, Y, O, \delta)$ は，状態集合 Q と出力記号集合 Y，出力関数 $O : Q \to Y$，状態遷移関数 $\delta : Q \times Q \to Q \times Q$ で構成される．Q と Y は有限集合である．O は各個体の出力を決定する．個体 v が状態 $q \in Q$ を持つとき，v の出力は $O(q)$ である．δ は，2 つの個体の交流が発生した際，交流後の各個体の状態を決定する．例として，状態 p, q, p', q' が $(p', q') = \delta(p, q)$ を満たすとする．このとき，状態 p を持つ個体 u と状態 q をもつ個体 v がそれぞれ呼びかけ側，応答側として交流を行った場合，各個体の交流後の状態はそれぞれ p', q' となる．

状況とは個体群中の全個体の状態を特定する写像 $C : V \to Q$ である．本稿では，プロトコル P について，個体群の取り得る全ての状況の集合を $\mathcal{C}_{\mathrm{all}}(P)$ と定義する．以下の条件を満たすとき，交流 $(u.v)$ によって状況 C から C' に遷移するといい，$C \xrightarrow{(u,v)} C'$ で表す．

$$(C'(u), C'(v)) = \delta(C(u), C(v))$$

$$C'(w) = C(w),\ w \in V \backslash \{u, v\}$$

これは，交流は各時刻において正確に 1 つだけ発生することを表す．実際のネットワークでは，交流がいつ発生するかわからず（非同期式モデル），複数の交流が同時に発生することもある．しかし，各個体は同時には 1 つの交流にしか参加できないとする限り，同時に発生する交流は 1 つだけとしても一般性は失われない．

各時刻 (=ステップ) において発生する交流は**スケジューラ**が決定する．一般に，個体群プロトコルを扱う研究では，一様ランダムなスケジューラを仮定する．つまり，各時刻において発生する交流は，発生し得る交流のうちから等確率かつ以前に発生した交流とは独立に選ばれる．本稿でも一様ランダムなスケジューラを仮定し，交流発生系列 $\Gamma = \Gamma_0, \Gamma_1 \dots$ で表す．ただし，$\Gamma_i \in E$ は $\Pr(\Gamma_i = (u, v)) = 1/n$ で表される確率変数とする．初期状況 C_0 および交流発生系列 Γ が与えられたとき，プロトコル P の実行は以下のように定義される．

$$\mathit{\Xi}_P(C_0, \Gamma) = C_0, C_1, \dots \quad \forall t (t \geq 0), C_t \xrightarrow{\Gamma_t} C_{t+1}$$

2.2 リーダ選挙問題

リーダ選挙問題は，分散システムにおける重要かつ基本的な問題である．個体群プロトコルモデルにおいては，一様ランダムなスケジューラによって交流が次々と発生していく結果として，個体群中のただひとつの個体をリーダ状態に，その他の個体を非リーダ状態にすることを目的とする．LE で示されるリーダ選挙の正当な挙動では，実行中，常に，リーダであるただひとつの個体 v が存在し，他のすべての個体は非リーダである必要がある．

期待維持交流回数 $EHT_P(C, LE)$ は，状況 $C \in \mathcal{C}_{\mathrm{all}}(P)$ から開始する実行 $\mathit{\Xi}_P(C)$ が LE を維持する交流の期待回数である．言い換えると，$\mathit{\Xi}_P(C)$ が LE から逸脱するまでの交流回数の期待値である．ただし，C が LE で定められた挙動を満たさない（リーダが存在しない，あるいは，複数個存在する）ときには，$EHT_P(C, LE) = 0$ と定義する．また，期待到達交流回数 $ECT_P(C, S)$ は，状況の集合 $S \subseteq \mathcal{C}_{\mathrm{all}}(P)$ について，状況 $C \in \mathcal{C}_{\mathrm{all}}(P)$ から開始する実行 $\mathit{\Xi}_P(C)$ が S 中の状況に到達するまでに要する交流回数の期待値である．

2.3 緩自己安定プロトコル

α, β を任意の正の実数とする．以下の式が成り立つ集合 S が存在するとき，プロトコル P は (α, β)-緩自己安定プロトコルであるという．

$$\max_{C \in \mathcal{C}_{\mathrm{all}}(P)} \ ECT_P(C, S) \leq \alpha$$

$$\min_{C \in S} \ EHT_P(C, LE) \geq \beta$$

直観的には，α が小さく，β が非常に大きい場合，(α, β)-緩自己安定プロトコルは有用である．

3 提案プロトコル

本節では，個体群の大きさが N 以下であるような任意のリンググラフにおける緩自己安定のリーダ選挙プロトコル P_{RL} を提案する．

本プロトコルでは，リーダを非リーダ変える役割を持つ**弾丸**とリーダが弾丸によって非リーダに変えられることを防ぐための**バリア**という考え方を用いる．以降では，P_{RL} の動作を擬似コードに従って説明する．個体 $v \in V$ の変数 `var` の値を v.`var` と表す．同様に，状態 $q \in Q$ の変数 `var` の値を q.`var` と表す．

Protocol 1 P_{RL}

Variables of each agent:

leader $\in \{\top, \bot\}$，bullet $\in \{\top, \bot\}$，barrier $\in \{\top, \bot\}$

remaining $\in [0, rem]$，$\mathrm{timer_L} \in [0, t_{\max}]$，$\mathrm{timer_B} \in [0, t_{\mathrm{bullet}}]$

Output function O**:**

if v.leader $= \top$ holds, then the output of agent v is l, otherwise f.

Interaction between initiator u_0 and responder u_1:

for all $i \in \{0, 1\}$ **do**
    if u_i.leader $= \bot$ **then**
        u_i.barrier $\leftarrow \bot$
    end if
end for

$u_0.\mathrm{timer_L} \leftarrow u_1.\mathrm{timer_L} \leftarrow \max(u_0.\mathrm{timer_L} - 1, u_1.\mathrm{timer_L} - 1, 0)$
if u_0.leader $= \top$ **or** u_1.leader $= \top$ **then**
    $u_0.\mathrm{timer_L} \leftarrow u_1.\mathrm{timer_L} \leftarrow t_{\max}$
else if $u_0.\mathrm{timer_L} = 0$ **or** $u_1.\mathrm{timer_L} = 0$ **then**
    $(u_0.\mathrm{leader}, u_0.\mathrm{timer_L}, u_0.\mathrm{timer_B}, u_1.\mathrm{timer_L}) \leftarrow (\top, t_{\max}, t_{\mathrm{bullet}}, t_{\max})$
end if

if u_0.bullet $= \top$ **and** u_0.remaining > 1 **then**
    $(u_1.\mathrm{bullet}, u_1.\mathrm{remaining}) \leftarrow (\top, u_0.\mathrm{remaining} - 1)$
end if
$(u_0.\mathrm{bullet}, u_0.\mathrm{remaining}) \leftarrow (\bot, 0)$
if u_1.leader $= \top$ **and** u_1.bullet $= \top$ **and** u_1.barrier $= \bot$ **then**
    u_1.leader $= \bot$
end if

for all $i \in \{0, 1\}$ **do**
    if u_i.leader $= \top$ **then**
        $u_i.\mathrm{timer_B} \leftarrow \max(u_i.\mathrm{timer_B} - 1, 0)$
    end if
end for
if u_0.leader $= \top$ **and** $u_0.\mathrm{timer_B} = 0$ **then**
    $(u_0.\mathrm{bullet}, u_0.\mathrm{barrier}, u_0.\mathrm{remaining}, u_0.\mathrm{timer_B}) \leftarrow (\top, \top, rem, t_{\mathrm{bullet}})$
end if
if u_1.leader $= \top$ **and** $u_1.\mathrm{timer_B} = 0$ **then**
    $(u_1.\mathrm{barrier}, u_1.\mathrm{timer_B}) \leftarrow (\bot, t_{\mathrm{bullet}})$
end if

Protocol 1 は，プロトコル P_{RL} の擬似コードである．擬似コードにおいて，$X \leftarrow Y$ は変数 X への値 Y の代入を表す．このプロトコルにおいて，各個体の状態は，3 つのバイナリ変数，$\texttt{leader} \in \{\top, \bot\}$，$\texttt{bullet} \in \{\top, \bot\}$，$\texttt{barrier} \in \{\top, \bot\}$ と，2 つのタイマ，$\texttt{timer}_{\mathrm{L}} \in [0, t_{\max}]$，$\texttt{timer}_{\mathrm{B}} \in [0, t_{\mathrm{bullet}}]$，0 以上の整数をとる 1 つの変数，$remaining \in [0, rem]$ の 6 つの変数から構成される．ただし，$t_{\max} = 28N$，$t_{\mathrm{bullet}} = 12N$，$rem = N$ とする．出力関数は，各個体について，$v.\texttt{leader} = \top$ である場合はリーダであることを示す記号 l を出力し，$v.\texttt{leader} = \bot$ の場合は非リーダであることを示す記号 f を出力する．

プロトコル P_{RL} は次の 4 つのパートから構成される．

- 調整パート（1〜5 行目）
- リーダ生成パート（6〜11 行目）
- 弾丸循環パート（12〜18 行目）
- 弾丸生成パート（19〜29 行目）

P_{RL} の目標は，任意の初期状況から個体群中でただひとつのリーダを選出することである． リーダ生成パートでは，リーダが個体群中に存在しない場合にリーダを生成する．他の 2 つのパートは連携して，リーダが 2 つ以上存在する場合にリーダの数を 1 つに減らす．以降では，呼びかけ側を x，応答側を y とする．

本プロトコルの実行中，新たにバリアを持つことができるのはリーダ個体のみである．しかしながら，初期状況では非リーダがバリアを持っている可能性がある．プロトコルの実行において，バリアを持つ非リーダが存在することは望ましくない状況である．そこで，調整パートでは非リーダが初めて参加した交流の最初にバリアを削除することで，各個体が望ましい状態を持つように調整する（1〜5 行目）．

リーダ生成パートの目的は，リーダが個体群中に存在しない場合にリーダを生成することである．各個体は，リーダが存在しない状況を検出するための指標として，リーダ生成タイマ $\texttt{timer}_{\mathrm{L}}$ を使用する．具体的には，x と y が交流する場合，$x.\texttt{timer}_{\mathrm{L}}$ と $y.\texttt{timer}_{\mathrm{L}}$ の大きい方の値から 1 減らした値を $x.\texttt{timer}_{\mathrm{L}}$ と $y.\texttt{timer}_{\mathrm{L}}$ に代入する（6 行目）．この交流には，**より大きなタイマの値を伝播させる**側面もある．x または y がリーダの場合，両個体のリーダ生成タイマは,$t_{\max}$ に初期化される（7，8 行目）．このイベントを**タイマリセット**と呼ぶ．個体が非リーダからリーダに変化するのは，リーダ生成タイマの値が 0 である非リーダ個体どうしで交流を行うとき，かつそのときのみである．このイベントを**タイムアウト**の発生という．タイムアウトが発生すると，P_{RL} は，個体群中にリーダ個体が存在しないものと見なし，x はリーダとなり，リーダ生成タイマの値を $t_{\max}$，弾丸生成タイマ $\texttt{timer}_{\mathrm{B}}$ の値を t_{bullet} に初期化する（9，10 行目）．リーダが存在しない状況では，タイマの初期化が発生しないため，タイムアウトは比較的短時間で発生する．少なくともひとつのリーダが存在する状況では，大きなタイマの値の伝播とタイマの初期化によって全個体のタイマが比較的大きな値に保たれるため，タイムアウトは滅多に発生しない．したがって，この仕組みは，ただひとつのリーダの安定性を実現する．

弾丸循環パートの目的は，個体群において，弾丸を循環させ，個体群中のリーダを削減することである．弾丸は寿命 $remaining$ を保持しており，寿命が 0 になるまで個体群中を循環し，特定の条件を満たすとき，リーダ個体を非リーダ個体に遷移させる．具体的には，x と y の交流において，寿命が 1 より大きい弾丸を x が持っている場合，x から y に弾丸を移動させる．その際，弾丸の寿命を 1 減らす．また，寿命が 1 である弾丸を x が持っている場合には，その弾丸は y に渡しても寿命が 0 になり役目を終えるため，y には渡さずに削除する（12〜15 行目）．ひとつの個体に弾丸が重なった場合には，後から来た弾丸の情報に更新される．リーダ y が弾丸を持っている場合について考える．y がバリアを持っていない場合，y は非リーダになる（16，17 行目）．バリアを持っている場合は，そのリーダが非リーダになることはない．いずれの場合も弾丸は巡回を続ける．弾丸はリングを順方向に 1 つ移動するとき，かつ，そのときのみ寿命を 1 減らすので，$rem = N$ であるから，一度生成されると，必ずリングを 1 周してから消滅する．

弾丸生成パートの目的は，弾丸を生成することである．弾丸を生成することができるのはリーダのみであり，弾丸を生成するかどうかを決定するための指標として弾丸生成タイマ $\texttt{timer}_{\mathrm{B}}$ を使用する．交流に参加するそれぞれの個体について，その個体がリーダである場合，弾丸生成タイマの値を 1 減ずる（20，21 行目）．x がリーダであり，かつ，弾丸生成タイマの値が 0 であるとき，x は，弾丸とバリアを生成し，弾丸生成タイマを初期化する（24，25 行目）．y がリーダであり，かつ，弾丸生成タイマの値が 0 であるとき，y は，バリアを失い，弾丸生成タイマを初期化する（27，28 行目）．すなわち，弾丸生成タイマが 0 となった個体は，次の交流で自身が呼びかけ側となったとき弾丸とバリアを生成し，応答側となったときバリアを失う．一様ランダムスケジューラの仮定から，各個体は，弾丸生成タイマが 0 になるたびに確率 1/2 で弾丸とバリアを生成し，残りの確率 1/2 でバリアを失うことになる．

プロトコル P_{RL} の実行は，任意の初期状況 C_0 から開始しても，やがてリーダの数をひとつにする．状況 C_0 においてリーダが存在しなければ，いずれタイムアウトが発生してリーダが生成される．リーダが複数存在する場合は，それぞれのリーダが弾丸生成タイマが 0 になったときに確率 1/2 で弾丸を生成することでリーダの数はやがて 1 になる．

リーダがただひとつになった後でも，リーダが存在しない，または複数のリーダが存在するという誤った状況になることがある．具体的には，リーダがただひとつ存在する状況でタイムアウトが起こると，複数のリーダが存在する状況になる．また，ただひとつのリーダが自身の弾丸によって殺されてしまう（自殺）と，リーダが存在しない状

況になる．しかしながら，このような事象は滅多に発生しない．リーダが存在するときは，前述の理由で，タイムアウトは滅多に発生しない．また，自殺についても，次に示す理由によって滅多に発生しない．自殺が起こる可能性があるのは，生成した弾丸が削除される前に，再度，弾丸生成タイマの値が 0 になり，バリアを失う場合である．弾丸生成タイマの値は，新しい弾丸を生成したときにのみ上限値 $\mathrm{timer_B}$ に初期化される．そのため，弾丸生成タイマの上限値 $\mathrm{timer_B}$ を弾丸の寿命 rem に比べ十分大きく設定していれば，再度，リーダの弾丸生成タイマの値が 0 になる前に，先に生成された弾丸は削除されるので，自殺はめったに起こらない．

上記の仕組みにより，提案プロトコルの実行は，任意の初期状況から実行を開始してやがてリーダがただひとつ存在し弾丸が存在しないような状況に到達し，その後（確率的に）極めて長い期間，ただひとつのリーダを維持する．

今，定理を示すための準備として，次のように集合を定義する．なお，簡単のため $\mathcal{C}_{\mathrm{all}}(P)$ を $\mathcal{C}_{\mathrm{all}}$ と表記する．

$$\mathcal{L}_{\mathrm{one}} = \{C \in \mathcal{C}_{\mathrm{all}} \mid \exists v_l \forall v \in V,\ C(v_l).\mathtt{leader} = \top \land (v \neq v_l \Rightarrow C(v).\mathtt{leader} = \bot)\}$$
$$\mathcal{T}_{\mathrm{half}} = \{C \in \mathcal{C}_{\mathrm{all}} \mid \forall v \in V,\ C(v).\mathtt{timer_L} \geq t_{\max}/2\}$$
$$\mathcal{B}_{\mathrm{zero}} = \{C \in \mathcal{C}_{\mathrm{all}} \mid \forall v \in V,\ C(v).\mathtt{bullet} = \bot\}$$
$$\mathcal{L}_{\mathrm{br}} = \{C \in \mathcal{C}_{\mathrm{all}} \mid \exists v \in V,\ C(v).\mathtt{leader} = \top \Rightarrow (C(v).\mathtt{barrier} = \top \land C(v).\mathtt{timer_B} \geq t_{\mathrm{bullet}}/2)\}$$
$$\mathcal{S}_{\mathrm{RL}} = \mathcal{L}_{\mathrm{one}} \cap \mathcal{T}_{\mathrm{half}} \cap (\mathcal{B}_{\mathrm{zero}} \cup \mathcal{L}_{\mathrm{br}})$$

このとき，提案プロトコルについて，このことを示す以下の 2 つの定理が成り立つ．

定理 1. $\mathcal{S}_{\mathrm{RL}}$ に属す任意の状況はプロトコル P_{RL} と挙動 LE に対する $\Omega(Ne^N)$-緩安全状況である．

定理 2. プロトコル P_{RL} は，挙動 LE と状況の集合 S_{RL} に対する $(O(nN\log n), \Omega(Ne^N))$-緩自己安定プロトコルである

余白の都合上，詳細な解析および証明は割愛するが，これらの定理から，このプロトコルの期待収束時間は $O(nN\log n)$ ステップであり，期待維持時間は $\Omega(Ne^N)$ ステップであることが示される．

4　まとめ

本稿では，個体群プロトコルモデルにおける緩自己安定リーダ選挙プロトコルを提案した．本プロトコルは，ネットワークサイズの上界が既知であるとの仮定のもと，リーダ選挙問題を有向リンググラフ上で緩自己安定的に解く．具体的には，提案プロトコルは，任意の初期状況から実行を開始して $O(nN\log n)$ 期待ステップで緩安全状況に到達し，それ以降，$\Omega(Ne^N)$ 期待ステップのあいだ唯一のリーダを保持し続ける．ただし，n はネットワークのサイズであり，N は n の既知の上界である．個体群プロトコルモデルにおいて，リーダ選挙問題を有向リンググラフ上で自己安定的に解くことができるプロトコルで現在提案されているものは安全状況に到達するために指数時間を要する．一方で，本プロトコルは緩安全状況に多項式時間で到達する．

今後の課題としては，期待収束時間及び期待維持時間を改善することや，ネットワークのサイズの上界が既知である仮定のもと，同一の問題について多項式時間で安全状況に到達する自己安定個体群プロトコルの提案等があげられる．

謝辞

研究の一部は JSPS 科研費 19H04085 および 20H04140 の助成を受けたものです．

参考文献

[1] 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):pp.235–253, 2006.

[2] Dana Angluin, James Aspnes, Michael J Fischer, and Hong Jiang. Self-stabilizing population protocols. *ACM Transactions on Autonomous and Adaptive Systems*, 3(4):Aritcle No.13, 2008.

[3] Hsueh-Ping Chen and Ho-Lin Chen. Self-stabilizing leader election. *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing*, pp.53–59, 2019.

[4] EW Dijkstra. Self-stabilizing systems in spite of distributed control. *CACM*, 17(11):pp.643–644, 1974.

[5] Yuichi Sudo, Junya Nakamura, Yukiko Yamauchi, Fukuhito Ooshita, Hirotsugu Kakugawa, and Toshimitsu Masuzawa. Loosely-stabilizing leader election in a population protocol model. *Theoretical Computer Science*, 444:pp.100–112, 2012.