 Vol.2016-AL-158 No.10
2016/6/25

# 個体群プロトコルにおける省スペースかつ高速な近似計数

江口 僚太[1,a)]　泉 泰介[1,b)]

**概要**：個体群プロトコルモデルとは $n$ 個のエージェント群が対で通信を繰り返すことで計算を行う分散システムのモデルであり，受動的モバイルネットワークの数理モデル化として良く知られている．本研究では，個体群プロトコルモデル上での近似計数問題，すなわちシステム中に存在するエージェントの総数 $n$ を近似的に推定する問題を考え，新たなアルゴリズムを提案する．提案アルゴリズムは，エージェントの総数への緩い上界 $N$ が与えられていることを仮定し，任意のエージェント対が等確率で通信を行う確率的スケジューラのもとで，計数問題に対して $O((\log\log N - \log\log(1+\epsilon))\log^2 N)$-時間，$O(1/\epsilon^2 \log 1/\epsilon + \log\log N - \log\log(1+\epsilon))$-ビットで $(1+\epsilon)$-近似値を出力する．なお，$(1+\epsilon)$-近似に必要となるビット数として $\log\log n$ ビットは最適である．

# Space and Time Efficient $(1+\epsilon)$-Approximation Counting on Population Protocols

***Abstract:*** Population protocol models is known as a model of passively mobile networks. It consists of $n$ agents and performs the distributed computation based on pairwise interactions between two agents. In this paper, we consider the approximate counting problem on the population protocol model, which requires the system to output an approximate value of $n$. The main contribution of this paper is to propose a new algorithm for approximate counting. It assumes a loose upper bound on $n$ (denoted by $N$), and for any $\epsilon > 0$ outputs $(1+\epsilon)$-approximate value of $n$ within $O((\log\log N - \log\log(1+\epsilon))\log^2 N)$parallel time. Since the proposed algorithm utilizes only $O(1/\epsilon^2 \log 1/\epsilon + \log\log N - \log\log(1+\epsilon))$-bit memory space for each agent, it achieves the optimal space complexity of the approximate counting problem.

## 1. はじめに

### 1.1 研究背景

近年，通信機器の小型化，軽量化に伴い，きわめて能力の制限された超小型計算機を用いた環境監視のためのネットワーク (センサネットワーク) が注目を集めている．本研究では特に，センサネットワークを構成する計算機群が何らかの移動体に取り付けられているようなシステム (モバイルセンサネットワーク) に注目する．モバイルセンサネットワークは小動物にセンサを取り付けた生態調査を行うなどのアプリケーションが考案されているが，来るべきサイバーフィジカル/IoT 時代において，その重要性は更に増しつつある．本稿では，モバイルセンサネットワークにおいて，個々の計算機を以降エージェントと呼ぶ．エージェント間の通信能力は限定されており，通信のための無線電波の可達範囲は非常に小さいことが常であるため，それらは十分に近接した時のみに通信 (インタラクション) を行うことが可能である．一般にエージェントは計算機（とその上のアルゴリズム）とは独立に移動を行い，どのエージェント対が近接するかどうかはシステム自体がコントロールすることはできない．このよう前提のもとでのモバイルネットワークを受動的モバイル（センサ）ネットワークと呼ぶ．本研究で扱う個体群プロトコルモデルは，2006 年に Angluin らによって提案された受動的モバイルセンサネットワークの 1 つのモデルであり，以下のような特徴を持つ [2].

- 通信は二者間通信を想定しており，どのエージェントが二者間通信を行うかは不定である．各エージェントの通信相手のスケジュールは，スケジューラと呼ばれる仮想的なデーモンにより制御される．スケジューラの動作をシステム自体が知ることはできず，また一般に生成されるスケジュールは非決定的である．

1 名古屋工業大学
Nagoya Institute of Technology Gokiso-cho, Showa-ku, Nagoya, Aichi, 466-8555 Japan
a) r.eguchi.816@nitech.jp
b) t-izumi@nitech.ac.jp

- 個体群を構成する各エージェントは，エージェントの総数を知らない．
- 個体群を構成するエージェントは互いに匿名である．つまり識別子を持たない．
- 各エージェントは，1 つの有限状態機械として振る舞う．この状態機械は，通信する相手の状態を外部からの入力として自身の状態遷移を行う．基本モデルでは個々のエージェントの状態数は定数であり，個体群の総数に依存しない．

具体的な定義は第 2 章で与えるが，個体群プロトコルモデルにおけるプロトコルとは，状態の集合とそれに対する状態遷移関数を決定するものとして規定される．

本研究は，個体群プロトコルモデルの基本的な問題である計数問題に対して対数時間 $(1+\epsilon)$-近似プロトコルを提案する（ここで $\epsilon$ とはアルゴリズムの設計パラメタである）．基本的なモデルでは状態数が個体群の総数 $n$ に非依存な定数であるという制限が課されているため，計数結果を格納するために十分なメモリ領域を確保できず，計数問題は自明に非可解である．そのため計数問題は，各エージェントのメモリ領域が定数でないように拡張されたモデル上で検討される [6]．本研究では，各エージェントはシステム内のエージェント総数に関しての緩い上界 $N$ に関してのの知識を持つことを仮定し，かつ，その状態数が $N$ に依存することを許す．ここで $N$ が緩い上界であるとは，ある定数 $c \geq 1$ に対して $N = O(n^c)$ であることを満たすことを言う．この事前知識の仮定から，本研究で検討する問題は，エージェント数に関する極めて緩い上界を $(1+\epsilon)$-近似値まで精度向上させる問題とみなすことができる．また，本研究では，次にインタラクションするエージェント対がその時点までのスケジュールとは独立に一様ランダムであることを仮定した確率的スケジューラを仮定する [1,6,7]．確率的スケジューラでは，プロトコルの実行時間は収束までにかかるインタラクション回数の期待値で測ることが可能であるが，実際のシステムにおいては，独立な 2 つのエージェント対のインタラクションは同時並行的に行われているとみなすことは自然である．そこで，実行における収束までのインタラクション回数の期待値を個体群の総数である $n$ で割った並行時間がしばしば時間的尺度として用いられる．なお，並行時間はインタラクションの総回数を単にシステムの時間不変な固定値 $n$ で割った値なので，総回数と並行時間の評価の間には本質的な差はない．以降本稿では単に「時間」と記載した場合は並行時間を指し，インタラクション回数は明示的に「回数」と述べる．

計数問題の本質的な難しさとしてしばしば挙げられることとして，個体群プロトコルにおいては各エージェントが固有の識別子を持たないという点がある．名前付けが行われていた場合，一般に計数問題は異なる識別子の個数を数え上げる問題と等価である．本研究ではプロトコル自身が乱択を用いることを許しているが，乱択を利用した場合，プロトコルの開始時に各エージェントが（十分に広い値域から）ランダムに識別子を設定することで，異なる識別子をもつシステムを高確率で模倣することができる．この設定においては識別子の有無は本質的な問題ではなくなる．しかしながら，個体群プロトコルモデル上での異なる識別子のカウントは，ナイーブな手法では $\Omega(n)$ 領域，$\Omega(n)$ 時間を必要とするため，極めて実現コストの高い手法となってしまう．本研究では，この問題をエージェントのサンプリングを用いることで解決する．提案アルゴリズムは，任意の定数 $\epsilon\ (0 < \epsilon < 1)$ に対して，$O(1/\epsilon^2 \log 1/\epsilon + \log\log N - \log\log(1+\epsilon))$ 領域，$O((\log\log N - \log\log(1+\epsilon))\log^2 N)$ 時間で高確率でエージェント数 $n$ の $(1+\epsilon)$-近似値 ($(1+\epsilon)^i \leq n \leq (1+\epsilon)^{i+1}$ を満たす整数 $i$) を計算する．$i$ の値の記録には明らかに $O(\log\log N)$ ビットを必要とするので，任意の $\epsilon < 1/\log\log n$ に対して，提案アルゴリズムは領域 (ビット数) 最適なアルゴリズムである．

### 1.2 構成

この項では論文の構成について述べる．1.3 項では関連研究について述べる．2 節では個体群プロトコルモデルと計数問題とその近似について定義を行っている．3 節では，$(1+\epsilon)$-近似を行う近似計数アルゴリズムの動作と解析を行っている．3.1 項では近似計数アルゴリズムの基本となる動作である判定アルゴリズムの動作を説明しており，その解析は 3.2 項に書かれている．3.3 項では近似計数アルゴリズムの全体の動作の説明と解析を示している．

### 1.3 関連研究

先述したとおり，個体群プロトコルモデルは 2006 年に Angluin らによって提案された．初出の論文，およびそれに引き続く一連の論文では，各エージェントがそれぞれ入力値を分散的に保持している状況において，個体群プロトコルの基本モデルによって計算することのできる述語について研究がおこなわれており，O(1) ビットの領域を持つエージェントで計算可能な入力データ上の述語のクラスが半線形述語と一致することが明らかになっている [3]．その後，いくつかの派生モデル，およびそれらの計算モデルについての検討がなされている [2,4]. 初期の研究に関するサーベイ論文としては [5] が挙げられる．

本稿が論じる計数問題は，個体群プロトコルモデルにおいては [6] においては初めて検討された．この論文では，全てのエージェント群と通信可能で潤沢な計算資源と計算能力を持つ基地局の存在を仮定している．同論文にとそれに引き続く論文において，基地局を含むモデルにおける計数問題の可解性，ならびにその領域複雑性が明らかにされている．

個体群プロトコルモデルにおける時間複雑性，特に確率

的スケジューラのもとでの準線形並列時間アルゴリズムについては近年リーダー選挙問題に関していくつかの進展がある．文献 [1] は各エージェントに対数多項式状態数を仮定することで，$O(\log^3 n)$ 時間でのリーダ選挙が可能なプロトコルを提案した．また，最近の結果として，エージェントの状態数が $O(1)$ であるような任意のプロトコルは確率的スケジューラのもとでの期待収束時間が $\Omega(n)$ 並列時間となることが示されている [7].

## 2. 諸定義

個体群プロトコルは $n$ 個のエージェントからなる．各エージェントは集合 $Q$ の要素を状態とする状態機械としてはたらく．状態機械が $\alpha$ ビット使用するとは，$2^\alpha$ 状態数を持つことと同値である．遷移関数は $\delta : Q \times Q \to Q \times Q$ であり，個体群プロトコルとは 2 項組 $(Q, \delta)$ である．また，エージェントのインタラクション可能性は自己ループを持たないインタラクショングラフ $G$ によって定められる．グラフ $G$ の頂点は個々エージェント，辺はエージェント間のインタラクションが可能なことを表す．多くの論文において，このグラフ $G$ は完全グラフを想定しており，本研究も同様である．プロトコルの実行はステップ単位で進められる．1 ステップで，グラフから一様ランダムに辺が選択されれ，その辺に対応するエージェント対 $(u, v)$ が遷移関数 $\delta$ により状態を更新する．

**定義 2.1 $\alpha$-近似計数問題** 個体群プロトコルが計数問題を $\alpha$-近似するとは，個体群内のエージェントの総数 $n$ と定数 $\alpha (> 1)$ に対してすべてのエージェントが, $\alpha^i \leq \hat{n} \leq \alpha^{i+1}$ を満たす近似値 $i$ を表す状態をもつよう収束することとする．

**定義 2.2 実行回数** 個体群内の全てのエージェントが高い確率 $1 - O(1/n^c)$ で目的の状態に収束するまでに必要なインタラクションの実行回数を，その個体群プロトコルの実行回数とする．

**定義 2.3 並行時間** 与えられた個体群プロトコルの実行回数を個体群の総数 $n$ で割った値をその実行の並行時間と呼ぶ．

## 3. $(1 + \epsilon)$ 近似計数アルゴリズム

提案する近似計数アルゴリズムのキーとなるアイデアは，任意の $i \in [0, \log N]$ によって定まる近似の候補 $\hat{n}_i = (1+\epsilon)^i$ について，システム内のエージェント数 $n$ が $n > \hat{n}_i$ かどうかを確率的に $O(\log N)$-時間で判定するアルゴリズムを構成することである．以下，3.1, 3.2 節でその判定アルゴリズムを示し，3.3 節ではそれを用いた近似計数アルゴリズムの全容について説明する．

### 3.1 判定アルゴリズム

擬似コードを **Algorithm1** に示す．判定アルゴリズムの動作の概要は，おおよそ以下のとおりである．

(1) 各エージェントは，初期状態において自身がアクティブかどうかを確率 $f(\epsilon)/\hat{n}_i$ のコイントスによって決定する．$f(\epsilon)$ は $\epsilon$ によって決まる定数であり，$f(\epsilon) = \frac{8(2+3\epsilon)^2}{\epsilon^2}$ である．

(2) アクティブなエージェント $x$ は $[1, r(\epsilon)]$ から一様ランダムに値を取得し，自身の ID とする ($r(\epsilon)$ がどのような値かについては後述)．そして，自身が保有する集合 $S_x$ に格納する．

(3) 2 つのエージェント $x, y$ がインタラクションしたとき両エージェントは両者が保持する値の集合の集合和を取る．すなわち，$S_x, S_y \leftarrow S_x \cup S_y$ とする．

(4) 各エージェントは $T_{out} = O(\log N)$ 回インタラクションを行ったのち，集合 $S_x$ の要素数がしきい値 $\sigma(\epsilon) = \lceil \frac{2}{2+3\epsilon} f(\epsilon) \rceil$ より大きければ $\hat{n}_i$ は $n$ 以下であると判断し，そうでなければ $n$ より大きいと判断する ($O(\log N)$ の具体的な値については後述する)．

この判定アルゴリズムが定数確率で正しく動作する理由を直感的に説明する．ステップ 1 のコイントスの段階において，アクティブなエージェントの個数の期待値は $f(\epsilon) \cdot n/\hat{n}_i$ であり，$\hat{n}_i$ が $n$ より小さいなら $f(\epsilon)$ より多く，$n$ の周辺ならば $f(\epsilon)$ の周辺すなわち定数であり，$n$ より大きいならば $f(\epsilon)$ より小さいことが期待できる．またもしステップ 2 の動作で，アクティブなエージェントが取得する ID の値がすべて異なるならば，それら全ての集合 $\bigcup_{i \in 1,n} S_i$ の要素数は計数に参加したエージェントの個数と一致する．この直感が定数確率以上で成功することは 3.2 節で示す．

次にこの判定アルゴリズムの収束時間が $O(\log N)$ 時間である理由について期待値の観点から説明する．まず，計数に参加するあるエージェントがアルゴリズムの実行前に取得した ID に注目し，それが他のエージェントに伝わっていく様子を観察する．説明の簡単のために，その ID は重複していないものとする．また，その ID を知らないエージェントを未知のエージェント，知っているエージェントを既知のエージェントと表記する．まず，あるエージェントがインタラクションに参加するまでにかかるステップ数の期待値は $n/2$ 回である．よって $O(n)$-ステップ経ったとき全てのエージェントは 1 回はインタラクションに参加しているとしていい．最初の $O(n)$-ステップでこのエージェントが他のエージェントとインタラクションしたとき，相手のエージェントは必ず未知のエージェントであり，既知のエージェントの総数が 1 増える．以降，クーポンコレクタの議論と同様にして，期待値の観点では全てのエージェントがその ID を取得するまでにかかるステップ数は $O(n \log n)$-ステップ (すなわち $O(\log n)$-時間) であることが示せる．判定アルゴリズム内においては複数のエージェントがアクティブとなり複数の ID をシステム内に散布し

ているが，それらのうち一つでも失敗する確率は和集合上界を用いることで上から十分に小さい定数で抑えることが出来る．すなわち，十分に大きい定数確率で全ての散布は完了する．

**Algorithm 1** 判定アルゴリズム

```
エージェント x が持つ変数：
    t_x：自身のインタラクション数カウンタ，0 で初期化
    S_x：ID を保存する集合
    result_x：結果を保持する変数
各エージェントに共通の定数：
    T_out ：エージェントがインタラクションする回数のタイムアウト値
    σ(ε)：計数のしきい値 (= ⌈16(2+3ε)/ε^2⌉)
    i：判定中の近似値の候補 n̂_i(= (1+ε)^i)
エージェント x の動作：
initilize()
  t_x ← 0
  if (coin_tosses(i) ==SUCCESS) then
      S_x ← {get_id()} //ID を取得して集合 S_x に格納
  endif
determine_n̂_i(i)
  when (x とある y がインタラクション) do
    if (t_x < ∞ ∧ t_y == ∞) then
        t_x ← ∞, return(result_x ← result_y)
    elseif(t_x < ∞ ∧ t_y < ∞) then
        t_x ← t_x + 1; S_x ← S_x ∪ S_y ; // 集合の和を取る
        if (count(S_x) ≥ σ(ε)) then
            return (result_x ← LESS), t_x ← ∞
        else
            if (t_x ≥ T_out) then
                return(result_x ← GREATER) , t_x ← ∞
            endif
        endif
    endif
```

### 3.2 判定アルゴリズムの解析

まず $O(\log N)$-時間後に 3/4 以上の確率で全てのエージェントが共通の集合 $S_x$ を持つことを示す．なお，あるアクティブなエージェントが自らの ID を他のエージェントに伝えていくことをそのエージェントによる ID の伝播と呼ぶことにする．

まず，(i)1 回の伝播が $O(\log n)$ 時間で収束すること，(ii) 任意のエージェントが $O(\log N)$ 時間経ったことを知ることができること (すなわち，自らのインタラクション数が $T_{out} = O(\log N)$ 回となったら高い確率で全体のステップ数が $\Omega(n \log n)$ であること) この 2 つを示すことで得られる．(i) は補題 3.1 から得られる．

**補題 3.1**　ある定数 $\xi(< n)$ について，$\xi$ 以上のエージェントが互いに異なる ID を持っているならば，$e^4\xi n \log n$ ステップ後に確率 $1 - 1/e^4$ 以上の確率で全てのエージェントが要素数 $\xi$ 以上の集合を保持している．

**証明**　まず ID の伝播を行うエージェントが 1 個のみの場合を考える．並行にに複数個のエージェントが伝播を行う場合は，これは和集合上界を用いることで導出することができる．

ある時点で当該 ID がすでに既知であるエージェントが $c(\geq 1)$ 個以上存在する場合を考える．未知のエージェントと既知のエージェントがインタラクションして既知のエージェントの数が増える確率は，$\frac{2c(n-c)}{n(n-1)}$ である．クーポンコレクタの議論と同様に，全てのエージェントが既知となるまでにかかるステップ数の期待値は，

$$\begin{aligned}\sum_{x=c}^{n-1} \frac{n(n-1)}{2x(n-x)} &\leq \sum_{x=1}^{n-1} \frac{n(n-1)}{2x(n-x)} \\ &\leq \frac{n-1}{2} \sum_{x=1}^{n-1} \left( \frac{1}{x} + \frac{1}{n-x} \right) \\ &\leq n \log n\end{aligned}$$

である．

Markov の不等式より，$e^4\xi n \log n$ ステップ後にその要素について全てのエージェントが既知でない確率は

$$\Pr\left(X \geq e^4 \xi n \log n\right) \leq \frac{1}{e^4 \xi}$$

であり，和集合上界をとって $\xi$ 個のアクティブなエージェントの互いに異なる全ての ID が全てのエージェントに $e^4\xi n \log n$ ステップ後に伝わらない確率は $1/e^4$ 以下である．

□

(ii) は補題 3.2 より得られる．補題 3.2 はあるエージェントが $O(\ln n)$ 回インタラクションしたとき高い確率で全体のステップ数が $\Omega(n \ln n)$ 回であることを示している．

**補題 3.2**　任意の $\beta > 3$ と時刻 $T$ について，$T + \beta n \ln n$ ステップ後に $1 - \frac{1}{n^2}$ の確率ですべてのエージェントのインタラクション数が $\alpha \ln n$ 回以下であるような，定数 $\alpha$ が存在する．

**証明**　あるエージェントがインタラクションに参加する確率は $\frac{2}{n}$ であり，これは時刻 $T$ とは独立である．よって $T$ から $\beta n \ln n$ ステップ中にそのエージェントがインタラクションに参加した回数は確率変数 $X \sim \mathrm{Bin}(\frac{2}{n}, \beta n \ln n)$ に従う ($\mathrm{Bin}(p, n)$ は試行回数 $n$，各試行の成功確率 $p$ の二項分布を表す)．この確率変数 $X$ に対して $\delta = \sqrt{\frac{3}{\beta}}$ とした Chernoff 上界を適用して

$$\begin{aligned}\Pr\left(X \geq (1+\delta)\mu\right) &\leq \exp\left( -\frac{1}{3} \cdot 2\beta \ln n \cdot \left( \sqrt{\frac{3}{\beta}} \right)^2 \right) \\ &\leq \exp\left(-2 \ln n\right) \leq \frac{1}{n^2}\end{aligned}$$

であり，$(1+\delta)\mu \leq \alpha n \ln n$ を満たす適切な $\alpha$ を決定することで証明が終了する．□

次に，このアルゴリズムが定数確率以上で正しい答えを返すことを示す．$\hat{n}_i = (1+\epsilon)^i$ とする．正当性の証明に必要な次の 2 つの事象が十分に高い確率で成立すること

Vol.2016-AL-158 No.10
2016/6/25

を示す．(1)$\hat{n}_i \leq n$ ならばアクティブなエージェントの個数が $\sigma(\epsilon)$ より大きくなり，そうでないならば小さくなる (2)$\hat{n}_i \leq n$ のときアクティブなエージェントが取得する異なる ID の個数が $\sigma(\epsilon)$ より多くなる．この 2 つはそれぞれ補題 3.3 と補題 3.4 で示される．

**補題 3.3** ある定数 $\epsilon < 1$ について以下を満たすような $\epsilon$ の関数 $f(\epsilon)$ と $\sigma(\epsilon)$ が存在する：任意の $\hat{n}_i$ について，各エージェントが確率 $\frac{f(\epsilon)}{\hat{n}_i}$ でコイントスするとき，$1-\frac{1}{e^4}$ 以上の確率で $\hat{n}_i \leq n$ ならばアクティブなエージェントの個数は $\sigma(\epsilon)$ 以上であり，$\hat{n}_i > n$ ならば $\sigma(\epsilon)$ 以下である．

**証明** $f(\epsilon) = \frac{8(2+3\epsilon)^2}{\epsilon^2}$，$\sigma(\epsilon) = \lceil \frac{16(2+3\epsilon)}{\epsilon^2} \rceil$ とする．また $d$ を $(1+\epsilon)^d \leq n < (1+\epsilon)^{d+1}$ を満たす定数とする．ある近似値の候補 $\hat{n}_i$ での判定アルゴリズムの実行において，アクティブなエージェントの個数を確率変数 $X$ とすると，$X$ はパラメータ $p_i = \frac{f(\epsilon)}{\hat{n}_i}$，$n$ の二項分布に従う．$X$ の期待値 $\mu_i$ は $\mu_i = \frac{f(\epsilon)n}{\hat{n}_i}$ となる．$\mu_i$ は $i$ の増加について単調減少であり，この補題を証明するためには境界条件 $\hat{n}_i = d$ と $\hat{n}_i = d+1$ の場合のみ考えれば良い事がわかる．
$i = d$ のとき，$\delta = \frac{\epsilon}{2+3\epsilon}$ として $X$ に Chernoff 上界を適用すると，

$$\begin{aligned}\Pr(X \leq (1-\delta)\mu) &\leq \exp\left(-\frac{8(2+3\epsilon)^2 n}{2\epsilon^2 \hat{n}_i} \cdot \frac{\epsilon^2}{(2+3\epsilon)^2}\right) \\ &\leq \exp\left(-4\frac{n}{(1+\epsilon)^i}\right) \leq \frac{1}{e^4}\end{aligned}$$

このとき以下が成立する．

$$\begin{aligned}(1-\delta)\mu &= \frac{2(1+\epsilon)}{2+3\epsilon} \cdot \frac{n}{\hat{n}_d} \cdot \frac{8(2+3\epsilon)^2}{\epsilon^2} \\ &\geq (1+\epsilon) \cdot \frac{16(2+3\epsilon)}{\epsilon^2} \geq \frac{16(2+3\epsilon)}{\epsilon^2} + 1 \\ &\geq \lceil \frac{16(2+3\epsilon)}{\epsilon^2} \rceil = \sigma(\epsilon)\end{aligned}$$

同様に，$i = d+1$ のとき $\delta' = \frac{3\epsilon}{2+3\epsilon}$ として Chernoff 上界を適用すると

$$\begin{aligned}\Pr(X \geq (1+\delta')\mu) &\leq \exp\left(-\frac{8(2+3\epsilon)^2 n}{3\epsilon^2 \hat{n}_{d+1}} \frac{9\epsilon^2}{(2+3\epsilon)^2}\right) \\ &\leq \exp\left(-24\frac{n}{(1+\epsilon)^{d+1}}\right) \\ &\leq \exp\left(-\frac{24}{1+\epsilon}\right) \leq \frac{1}{e^{12}}\end{aligned}$$

ここで，以下が成立する．

$$\begin{aligned}(1+\delta')\mu &= \frac{2(1+3\epsilon)}{2+3\epsilon} \cdot \frac{n}{\hat{n}_{d+1}} \cdot \frac{6(2+\epsilon)^2}{\epsilon^2} \\ &\leq (1+3\epsilon) \cdot \frac{12(2+\epsilon)}{\epsilon^2} \\ &\leq \lceil 24(2+\epsilon)(1+\epsilon)/\epsilon^2 \rceil = \sigma(\epsilon)\end{aligned}$$

以上より補題 3.3 が証明される． □
補題 3.4 はアクティブなエージェントの個数が $\sigma(\epsilon)$ 以上だったとき，一様ランダムに取得する ID の値域が $[1, O(\sigma^2(\epsilon))]$ であれば定数確率で計数が正しく行われることを示す．すなわち，アクティブなエージェントの個数 $m$ が $m \geq \sigma(\epsilon)$ であったとき異なる ID を持つエージェントの個数が $\sigma(\epsilon)$ より小さくなる確率は $1/e^4$ 以下であることを示す．

**補題 3.4** $\sigma(\epsilon) = \lceil \frac{16(2+3\epsilon)}{\epsilon^2} \rceil$，$r = \lceil e^4(\sigma(\epsilon)-1)^2 \rceil$ に対し，アクティブなエージェントは値域 $[1, r]$ から一様ランダムに値を取得するとする．このとき，$m(\geq \sigma(\epsilon))$ 個以上のエージェントがアクティブであるなら，異なる ID の個数は $1 - 1/e^4$ 以上の確率で $\sigma(\epsilon)$ 以上となる．

**証明** $m = \sigma(\epsilon)$ のときを考える．一般性を失うことなく，各エージェントが順番に値を取得すると考えることができ，そのとき全ての ID が異なる確率は

$$\begin{aligned}\prod_{i=1}^{\sigma(\epsilon)-1}\left(1-\frac{i}{k}\right) &\geq \left(1-\frac{(\sigma(\epsilon)-1)}{e^4(\sigma(\epsilon)-1)^2}\right)^{\sigma(\epsilon)-1} \\ &\geq \left(1-\frac{(\sigma(\epsilon)-1)^2}{e^4(\sigma(\epsilon)-1)^2}\right) = 1-\frac{1}{e^4}\end{aligned}$$

である．よって $m = \sigma(\epsilon)$ のとき異なる ID の個数は $1-1/e^4$ 以上の確率で $\sigma(\epsilon)$ 以上となる．
つぎに $m > \sigma(\epsilon)$ のときを考える．このときもまた一般性を失うことなく各エージェントが順番に値を取得すると考えることができる．最初の $\sigma(\epsilon)$ 個のエージェントが取得する ID がすべて異なる確率は，$m = \sigma(\epsilon)$ のときと同様にして $1 - 1/e^4$ 以上の確率であり，よって $m > \sigma(\epsilon)$ のときもまた異なる ID の個数は $1 - 1/e^4$ 以上の確率で $\sigma(\epsilon)$ 以上となる．以上より補題が証明される． □

最後に，この判定アルゴリズムの解析をまとめる．

**定理 3.5** この判定アルゴリズムは，ある近似値の候補 $\hat{n}_i = (1+\epsilon)^i$ に対して，確率 $1-\frac{1}{e^2}$ 以上で $\hat{n}_i \leq n$ かどうか判定する．使用するビットは $O(\frac{1}{\epsilon}^2 \log\frac{1}{\epsilon} + \log\log N)$-ビットであり，$O(\log N)$-時間で終了する．

**証明** $\hat{n}_i \leq n$ である場合を考える．補題 3.3 より，アクティブなエージェントの個数は $1-\frac{1}{e^4}$ の確率で $\sigma(\epsilon)$ より多い．また，補題 3.4 より確率 $1-\frac{1}{e^4}$ 以上でアクティブなエージェントが持つ異なる ID の個数は $\sigma(\epsilon)$ である．このとき，補題 3.1 より $1-\frac{1}{e^4}$ 以上で $O(\log N)$-時間で全てのエージェントが要素数 $\sigma(\epsilon)$ 以上の集合を保持しており，各エージェントが計数した結果は正しくなる．使用しているビットは ID を格納するために使用する $O(\frac{1}{\epsilon^2}\log\frac{1}{\epsilon})$-ビットと，自身のインタラクション回数を数えるカウンタが使用する $O(\log\log N)$-ビットである．成功確率は失敗確率について和集合上界をとって $1-\frac{1}{e^2}$ 以上となる．
次に $\hat{n}_i > n$ である場合を考える．補題 3.3 より $1-\frac{1}{e^12}$ 以上の確率でアクティブなエージェントの個数は $\sigma(\epsilon)$ より小さい．補題 3.2 よりあるエージェント $O(\log N)$ 時間後

にタイムアウトしたことを知り，そのエージェントは実行が終了したことを他のエージェントに伝えていく．これもまた伝播であり，$O(\log N)$ 時間で終了する．結果的に $O(\log N)$-時間で実行は終了する．成功確率は失敗確率について和集合をとって $1-\frac{1}{e^2}$ 以上となる．

### 3.3 近似計数アルゴリズム

判定アルゴリズムを基に作られる $(1+\epsilon)$-近似を達成する近似計数アルゴリズムを **Algorithm2** に示す．次に近似計数アルゴリズムの説明をする．このアルゴリズムに対する形式的な解析は定理 3.6 に示す．

判定アルゴリズムの成功確率は，$1-1/e^2$ 以上であった．これを高確率で成功させるために，計数近似アルゴリズムでは同じ近似値の候補 $\hat{n}_i$ に対して $O(\log N)$ 回判定アルゴリズムを実行し，多数決を取ることでその $\hat{n}_i \le n$ かどうか判定する（**Algorithm2** における 1-6 行目の while ループ部分に相当）．なお，**Algorithm2** には他のエージェントとインタラクションした時の動作が書かれていないが，それは関数 $determine\_\hat{n}_i()$ の内部のみで行われている．すなわち，**Algorithm2** は，判定アルゴリズムが終了したのち，同じ候補でもう一度繰り返すのか，もしくは他の候補に移るのか，それとも計算を終了するのか，この 3 つを決定している補助的な処理とみなすことができる．

**Algorithm 2** 近似計数アルゴリズム

エージェント $x$ が持つ変数：
- $i_x$：判定中の $i$ 番目の近似値の候補 $\hat{n}_i$
- $r_x$：自身のラウンド数カウンタ，0 で初期化
- $less_x, greater_x$：カウンティング結果のカウンタ
- $head_x, tail_x$：近似値の候補の範囲

各エージェントに共有の定数：
- $R_{out}$ ：ラウンド数の上限

エージェント $x$ の動作：

```
1: while(head_x ≠ tail_x) do
2:     if(determine_n̂_i() ==LESS) then less_x++
3:         else greater_x++
4:     endif
5:     next_round()
6: endwhile
7: return(head_x)
```

補助プロシージャ $next\_round()$：

```
1: if (r_x ≥ R_out) then
2:     if (less_x ≥ greater_x) then
3:         head_x ← i_x + 1
4:     then
5:         tail_x ← i_x − 1
6:     endif
7:     i_x ← ⌈(head_x+tail_x)/2⌉ //次の候補を決定
8: else
9:         r_x ← r_x + 1
10: endif
11: initilize(i)
```

後に示すように，$O(\log N)$ 回繰り返せば，同じ近似値の候補 $\hat{n}_i \le n$ の判定が高確率で成功する．このアルゴリズムでは，判定が成功した後，$\hat{n}_i > n$ ならば $\hat{n}_i$ より前の候補に対してまた判定アルゴリズムを実行し，$\hat{n}_i \le n$ ならば $\hat{n}_i$ より後の候補に対して判定アルゴリズムを実行する，二分探索アルゴリズムを実行している．近似値の候補 $\hat{n}_i$ は $\log_{(1+\epsilon)} N$ 個存在している．結果的にこのアルゴリズムは，判定アルゴリズムの 1 回の実行が $\log N$ 時間で，それを $\log N$ 回，最終的に $\log\log_{(1+\epsilon)} N$ の近似値の候補に対して行うので実行時間は $O((\log\log N - \log\log(1+\epsilon))\log^2 N)$ 時間となる．使用するビットは，アクティブなエージェントの ID を保存するための $O(1/\epsilon^2 \log 1/\epsilon)$ ビットと近似値の候補を保存しておくための $O(\log\log_{(1+\epsilon)} N)$ ビット，合計で $O(1/\epsilon^2 \log 1/\epsilon + \log\log N - \log\log(1+\epsilon))$ ビット必要となる．

**定理 3.6** ある定数 $\alpha > 0$ について，判定アルゴリズムを一つの近似値の候補に対して $\alpha e^4 \log N$ 回実行するとき，近似計数アルゴリズムは $1-1/n$ 以上の確率でエージェントの総数 $n$ に対する $(1+\epsilon)$ 近似値を返すような，$\alpha$ が存在する．

**証明** 二分探索の構造から，$\log_{(1+\epsilon)} N$ 個の近似値の候補から $\log\log_{(1+\epsilon)} N$ 個の候補を判定すれば十分であることは明らかである．以降では一つの候補に対する成功確率が $1-1/n^2$ であることを示す．定理 3.6 は，これに対して和集合上界を取ることで得られる．

ある整数 $\beta > 0$ を $\log\log N - \log\log(1+\epsilon) \le N^\beta$ を満たす値とする．$\alpha = 12(\beta+1)$ とする．判定アルゴリズムの成功確率は，任意の近似値の候補 $\hat{n}_i$ に対して $1-1/e^2$ であった．アルゴリズムの失敗回数が任意の近似値の候補 $\hat{n}_i$ に対して $12(\beta+1)e^4 \log N$ 回実行したとき $6(\beta+1)e^4 \log N$ 回以上失敗する確率が $1/N^{\beta+1}$ 以下であることを示す．ここで判定アルゴリズムの失敗とは，$\hat{n}_i \le n$ であるとき全てのエージェントが $\hat{n}_i > n$ と判定することとする．

失敗回数 $Z$ は二項分布 $\mathrm{Bin}(\frac{1}{e^4}, 12(\beta+1)e^4 \log N)$ に従う．これに Chernoff 上界を適用して，

$$\begin{aligned}\Pr(Z \ge 18(\beta+1)\log N) &= \Pr\left(Z \ge \frac{3}{2}12(\beta+1)\log N\right) \\ &\le \exp\left(-\frac{1}{3}12(\beta+1)\log N \frac{1}{2^2}\right) \\ &\le \frac{1}{N^{\beta+1}}\end{aligned}$$

これに対して，$\beta$ 回和集合上界を適用することで，定理の言明を得ることができる． □

### 参考文献

[1] Alistarh, D. and Gelashvili, R.: Polylogarithmic-Time Leader Election in Population Protocols Using Polylogarithmic States, *arXiv preprint arXiv:1502.05745* (2015).

[2] Angluin, D., Aspnes, J., Diamadi, Z., Fischer, M. J. and Peralta, R.: Computation in networks of passively mobile finite-state sensors, *Distributed computing*, Vol. 18, No. 4,

pp. 235–253 (2006).

[3] Angluin, D., Aspnes, J. and Eisenstat, D.: Stably computable predicates are semilinear, *Proceedings of the twenty-fifth annual ACM symposium on Principles of distributed computing*, pp. 292–299 (2006).

[4] Angluin, D., Aspnes, J., Eisenstat, D. and Ruppert, E.: The computational power of population protocols, *Distributed Computing*, Vol. 20, No. 4, pp. 279–304 (2007).

[5] Aspnes, J. and Ruppert, E.: An introduction to population protocols, *Middleware for Network Eccentric and Mobile Applications*, Springer, pp. 97–120 (2009).

[6] Beauquier, J., Clement, J., Messika, S., Rosaz, L. and Rozoy, B.: Self-stabilizing counting in mobile sensor networks with a base station, *Distributed Computing*, Springer, pp. 63–76 (2007).

[7] Doty, D. and Soloveichik, D.: Stable leader election in population protocols requires linear time, *arXiv preprint arXiv:1502.04246* (2015).

[8] Mitzenmachr, M. and Upfal, E.: 確率と計算一乱択アルゴリズムと確率的解析一，chapter 1–5, 共立出版 (2009).