

# 離散格子平面におけるライト付き自律分散ロボット群の 完全相互観測アルゴリズム

石田 湧之<sup>†</sup> 金 鎔煥<sup>†</sup> 片山 喜章<sup>†</sup>

<sup>†</sup>名古屋工業大学大学院工学研究科情報工学専攻

E-mail: <sup>†</sup>yuuno@moss.elcom.nitech.ac.jp, <sup>††</sup>{kim,katayama}@nitech.ac.jp

あらまし 自律分散ロボットシステムとは複数の移動可能な端末（以下、ロボット）で構成される分散システムの一つで、各ロボットは「他ロボットの位置の観測 (Look)」 「移動先の計算 (Compute)」 「移動 (Move)」 のサイクルを繰り返すことで与えられた問題を解決する。自律分散ロボットに関する研究の多くは、集合問題や形状形成問題など様々な問題を想定し、与えられた問題を解決するために必要なロボットの機能群 (仮定している能力) を理論的に解明することをその目的としている。本研究では、離散格子平面上に任意に配置されているロボット群において、全てのロボットがお互いを観測できる位置に再配置する完全相互観測 (Complete Visibility) 問題を扱う。各ロボットは透過性を持たないため、全てのロボットがお互いを観測できるためには適切な配置が必要となる。また、ロボットのモデルにはライトを持つものもあり、ライト付きの自律分散ロボット群をルミナスモデルという。本論文では、2種類の観測可能なライトを有し、完全非同期で動作するロボット群において、ロボットの総台数に関する知識を持たない場合でも正しく完全相互観測問題を解決するアルゴリズムを提案する。

キーワード 自律分散ロボットシステム, ルミナスモデル, 完全相互観測問題

## On a Complete Visibility for Autonomous Mobile Robots with Lights in Asynchronous scheduler on a Grid Plane

Yuno ISHIDA<sup>†</sup>, Yonghwan KIM<sup>†</sup>, and Yoshiaki KATAYAMA<sup>†</sup>

<sup>†</sup> Nagoya Institute of Technology, Graduate School of Computer Science and Engineering

E-mail: <sup>†</sup>yuuno@moss.elcom.nitech.ac.jp, <sup>††</sup>{kim,katayama}@nitech.ac.jp

**Abstract** Autonomous mobile robot system is a distributed system consisting of autonomous mobile computational entities (called robots) which operate in Look-Compute-Move cycles. Many problems for autonomous mobile robots, e.g., gathering, pattern formation or flocking, are considered and many studies have been presented to clarify the relation between the computational capability of each robot and the solvability of the problem. We consider the complete visibility problem which is a problem to relocate all the robots on a grid plane so that each robot is visible to all others. We assume that each robot has two-colored light, i.e., a persistent value which is visible by the other robots, is opaque, i.e., not transparent, and has unlimited visibility range. In this paper, we propose an algorithm to achieve a complete visibility when a set of autonomous mobile robots is given. The proposed algorithm guarantees a complete visibility even if every robot operates asynchronously and no robot knows the total number of the robots on the grid plane.

**Key words** autonomous mobile robot system, luminous model, Complete Visibility problem

### 1. ま え が き

自律移動ロボットシステム [1] とは、低機能で自律的に移動可能な複数の端末によって構成されるシステムである。端末の機能やシステム全体としての仮定の違いによってさまざまなモデルが考えられる。そのモデルのわずかな違いでさえ、システ

ムによる問題の可解性に大きく作用する点が興味深く、特に分散システムの研究者を中心として盛んに研究されている。自律移動ロボットの分野では様々な研究が行われており、例えば、形状形成 [2] や一点集合 [3] などが挙げられる。

自律分散ロボットを特徴づけるモデルの一つに、ロボットの透過性がある。ロボットが透明なモデルでは、観測時に視野の

妨げにならない。一方、不透明のモデルでは、2 台のロボットを結ぶ線分上に他のロボットが存在した場合、2 台のロボットはお互いを観測することができない。不透明なモデルにおける自律分散ロボットの問題として、完全相互観測問題 [4] があり、これは全てのロボットがお互いを観測できるように衝突を回避しながら再配置させる問題である。衝突とは、既にロボットの存在する点に他のロボットが移動しようとする、ロボット同士がすれ違うように移動しようとする、複数のロボットが同時に同じ点に移動しようとする状況を指す。本研究では、完全相互観測問題を解くアルゴリズムを考えるにあたってルミナスモデルを用いる [5]。ルミナスモデルとは、それぞれのロボットがあらかじめ決められた色の絶え間ない光を発することのできるライトを備えているモデルである。ルミナスモデルはライトのモデルの違いによって、大きく分けて 3 種類のモデルが存在する [6]。full-light とは、各ロボットのライトが自分自身と自分以外の他のロボットから観測可能なモデルである。external-light とは、各ロボットのライトが自分以外のロボットから観測可能で、自分からは観測不可能であり、ライトの色は自分で変更できるモデルである。internal-light とは、各ロボットのライトが自分からは観測できるが、自分以外の他のロボットからは観測できないモデルである。ルミナスモデルでは、各ロボットは他のロボットの位置と色を観測 (Look) し、観測結果を入力にアルゴリズムに従って行き先と次のサイクルの自身の色を計算 (Compute) して自身の色を変える、その行き先に移動 (Move) するというサイクルになる。

本研究では、完全非同期 (ASYNCR) に動作するルミナスモデルの自律分散ロボットによる格子平面上での完全相互観測問題を解くアルゴリズムを提案する。

## 2. モデルと諸定義

### 2.1 モデル

[座標系]: ロボットが存在する平面は格子平面  $\mathbb{T}$  とする。ロボットは共通座標系を持たず、 $x$  軸・ $y$  軸に合意を持つ。

[ロボット]: ロボットの総台数  $N$  となる自律分散ロボット群の集合を  $\mathcal{R}=\{r_1, r_2, \dots, r_N\}$  とする。これらのロボットは識別子を持たず、外見上区別することができない。また、ロボットは内部メモリを持たず、自分自身の過去の情報を利用できず、ロボットの総台数  $N$  を知らない。

[ロボットの移動]: ロボットは一回の移動で、自身が存在している格子点に隣接している 6 つの格子点の 1 つにのみ瞬間的に移動することが可能である。今回は完全非同期スケジューラの下にロボットの動きが制御されるため、移動中で格子点以外に存在するロボットを他のロボットが観測してしまうことはない。

[ライト]: それぞれのロボットは他のロボットから視覚可能な、予め用意された種類の色を発するライトを備えている。ロボットのライトは絶え間なく途切れることはない。今回提案するアルゴリズムで用いるライトの色は  $C \in \{A, B\}$  の 2 色だけである。

[視野]: ロボットの視野範囲に制限はない、つまり視野半径は無限と仮定する。しかし、ロボットに透過性はなく、ロ

ボット  $r_i$  と  $r_j$  を結ぶ線分上にロボット  $r_k$  が存在した場合、ロボット  $r_i$  と  $r_j$  はお互いを観測することができない。ロボット  $r$  はある時刻  $t$  において観測できた  $M$  台のロボットを  $r_0, r_1, \dots, r_M$  として、それぞれのロボットのライトの色と座標を得る。ある任意のロボット  $r_i (0 \leq i \leq M)$  のライトの色と座標を局所座標系を用いて、 $r_i = (x_i, y_i, Col_i) ((x_i, y_i) \in r)$  を中心とした格子点の座標、 $Col_i \in \{A, B\}$  と表現する。そして、ある時刻  $t$  におけるロボット  $r$  によって観測されるロボットの集合を  $\mathcal{V}_r(t) = \{r_0, r_1, \dots, r_M\}$  と表現する。

[Look-Compute-Move サイクル]: 各ロボットの動作は次に説明する 4 つの状態 *Wait*, *Look*, *Compute*, *Move* のうち後者の 3 つの状態を 1 サイクル (LCM サイクルと呼ぶ) とし繰り返し実行する。

**Wait:** ロボットは待機状態。ロボットは無制限に待機することはできない。開始時点ではすべてのロボットが *Wait* 状態である。

**Look:** ロボットは自身のライトの色、及び自身を中心とする座標系における他のロボットの位置座標とライトの色を得る。*Look* の結果、ロボットは局所座標系における観測したロボットの座標とライトの色の集合  $\mathcal{V}_r = \{r_0, r_1, \dots, r_M\}$  を得る。

**Compute:** ロボットは、*Look* で得られた  $\mathcal{V}_r$  のみを入力としてアルゴリズムを実行し、次のライトの色を決定し、移動先座標を求める。

**Move:** 色を変えた直後に *Compute* で求めた移動先座標に向かって移動する。

[スケジューラ]: ロボットは LCM サイクルをスケジューラから指示されたタイミングで実行すると仮定する。スケジューラは、その同期の程度によって次の 3 つのモデルが定義される (図 1)。本論文ではロボットは ASYNCR で動作するものとする。

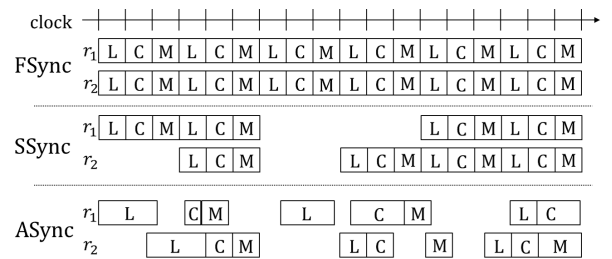


図 1 ロボットの実行モデル FSYNC,SSYNC,ASYNCR の実行例。図中の L, C, M はそれぞれ L:Look, C:Compute, M:Move を表している。モデルによりサイクルを開始するタイミングと開始するロボット、および各状態にかかる時間が異なる。

- **完全同期 (fully-synchronous, FSYNC):** すべてのロボットが完全に同期した時計を用いてアルゴリズムを実行できると仮定する。すなわち、すべてのロボットは同じ時刻にアルゴリズムを開始し、その後、同じ時刻に、*Look*, *Compute*, *Move* をそれぞれ実行する。

- **半同期 (semi-synchronous, SSYNC):** FSYNC において、サイクルを実行しないロボットの存在を許すモデルである。ただし、すべてのロボットは無限回サイクルを実行すると仮定する。

- 非同期 (asynchronous, ASYNC): ロボットの実行に関する仮定を一切置かないモデルである。各ロボットのサイクルの開始時刻や *Look*, *Compute*, *Move* にかかる時間の上限も与えられない。そのため、ロボットが移動中に他のロボットに観測されることもある。

これらの3つの同期モデルに関して、SSYNC は FSYNC の実行を含み、ASYNC は SSYNC, FSYNC の実行を含む。つまり、ASYNC である問題を解くアルゴリズムが存在する場合、そのアルゴリズムは SSYNC, FSYNC のロボットに対しても、その問題を正しく解くアルゴリズムである SSYNC と FSYNC のモデル関係も同様である。

## 2.2 完全相互観測問題の定義

本研究では自律分散ロボットシステムのルミナスモデルにおいて、“格子平面において、総台数  $N$  台のロボットがライトの色 ‘A’ で任意の格子点上に配置されている状況”を入力として、“格子平面において、総台数  $N$  台のロボットのライトの色が ‘B’ で格子点上に存在して、任意の3台のロボットが一直線上に存在しない状況”を出力として定義する。

## 2.3 諸 定 義

ロボット  $r$  を原点とする格子座標系において、各象限と軸上を次のように表現する。

[各象限]:  $Q_i (0 \leq i \leq 4)$

[プラス側・マイナス側の  $x$  軸上]:  $x_+, x_-$

[プラス側・マイナス側の  $y$  軸上]:  $y_+, y_-$

ある時刻  $t$  におけるロボット  $r$  自身のライトの色を返す関数を次のように定義する。

$C_r(t)$

例:

(1)  $C_r(t) == 'A'$

ロボット  $r$  の時刻  $t$  におけるライトの色は ‘A’ である。

ある時刻  $t$  におけるロボット  $r$  の観測結果を  $\mathcal{V}_r(t) = \{r_0, r_1, \dots, r_M\}$  として、各象限や軸上、座標上に存在するロボットの個数を返す関数を次のように定義する。

$\mathcal{V}_{r,p,C}(t) (p \in \{Q_i, x_+, x_-, y_+, y_-, (x, y)\}, C \in \{A, B\})$

例:

(1)  $\mathcal{V}_{r-Q_1,A}(t) == 3$

ロボット  $r$  は第一象限にライトの色 ‘A’ のロボットを3台観測する。

(2)  $\mathcal{V}_{r-x_-,B}(t) == 0$

ロボット  $r$  は  $x$  軸上マイナス方向にライトの色 ‘B’ のロボットを観測しない。

また、観測結果によって決定した移動先とライトの色を次のように表す。移動する軸を  $a \in \{x, y\}$ , 方向を  $d \in \{+, -\}$ , ライトの色を  $C \in \{A, B\}$  として  $(a^d, C)$  と表現する。0 は移動や色の変更はしない。

## 3. 提案アルゴリズム

### 3.1 アルゴリズムの基本戦略

本章では、完全非同期スケジューラにおける完全相互観測問題を解くアルゴリズムを説明する。

提案アルゴリズムではまず任意の位置に配置されたロボットを  $y$  軸上に一直線に隙間なく並べることを考える。これを直線形成フェーズと呼ぶ。次に、完全相互観測を達成するための基準点を定めることを考える。これは、これはグローバル座標系を持たないロボットが共通の座標系を得ると同時に、ロボットの総台数  $N$  を知るために必要である。このフェーズを座標系生成フェーズと呼ぶ。最後に、ロボットは基準点を観測することによって生成された座標系を計算し、完全相互観測を達成する位置へ移動する。このフェーズを完全相互観測達成フェーズと呼ぶ。座標系生成フェーズ以降はロボットの台数  $N$  が2台, 3台, 4台以上の場合で違うアルゴリズムを実行するため、場合分けをしなければならない。各フェーズの詳細を次に述べる。

### 3.2 直線形成フェーズ

このフェーズにおける次のようなロボットの配置状況をそれぞれ状況 A, B, C, D と定める。

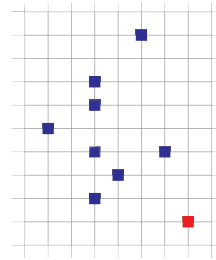


図2 状況 A

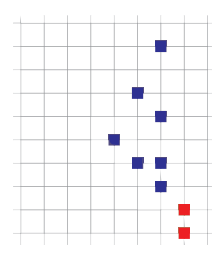


図3 状況 B

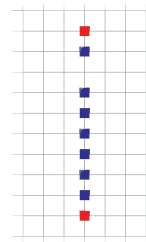


図4 状況 C

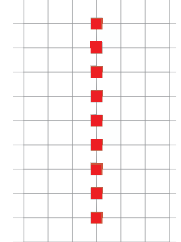


図5 状況 D

状況 A 1台のロボットのみがライトの色 ‘B’ であり、他の全てのライトの色 ‘A’ のロボットより  $y$  座標値が小さく、 $x$  座標値が大きい格子点に位置している。

状況 B 2台のロボットのみがライトの色 ‘B’ であり、同一  $y$  軸上のお互いが観測できる格子点上に存在している。

状況 C 全てのロボットは同一の  $y$  軸上に位置していて、 $y$  軸値が最も小さいロボットと  $y$  軸値が最も大きいロボットのみがライトの色 ‘B’ であり、他のロボットの色は ‘A’ である。

状況 D 全てのロボットは同一の  $y$  軸上に位置していて、全てのロボットはがライトの色 ‘B’ であり、隙間なく並んでいる。

始めに、初期配置から直線を形成するための基準点を定める。基準点上以外のロボットは基準点と同じ  $y$  軸上に直線を形成する。初期配置において、全てのロボットはライトの色 ‘A’ であるので、その中で第三象限、第四象限、同一  $y$  軸のマイナス方向、同一  $x$  軸のプラス方向にロボットを観測できないロボット

を基準ロボット  $\alpha$  として、基準点まで移動させることを考える。ロボット  $\alpha$  は同一  $x$  軸のマイナス方向にロボットを観測できたならば、 $y$  軸に沿ってマイナス方向に移動し(図??), 観測できないならば、 $x$  軸に沿ってプラス方向へ移動する(図 7)。ロボット  $\alpha$  は同一  $x$  軸のマイナス方向と第一象限にロボットを観測できないならば、基準点へ到達したとしてライトの色を‘ $B$ ’に変更する(図 8)。 $\alpha$  ロボットがライトの色を‘ $B$ ’に変更したとき、状況は状況 A となる。

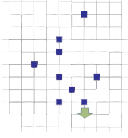


図 6 Rule1

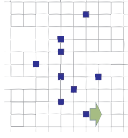


図 7 Rule2

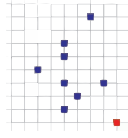


図 8 Rule3

---

**Algorithm 1** 直線形成フェーズ-初期配置から状況 A

---

時刻  $t$  におけるロボット  $r_i \in \mathcal{R}$  の観測結果を  $\mathcal{V}_{r_i}(t)$  とする。

**Rule1:**  $C_{r_i} == 'A' \wedge \mathcal{V}_{r_i\_any,B}(t) == 0 \wedge \mathcal{V}_{r_i\_Q3,A}(t) == 0 \wedge \mathcal{V}_{r_i\_Q4,A}(t) == 0 \wedge \mathcal{V}_{r_i\_y-,A}(t) == 0 \wedge \mathcal{V}_{r_i\_x+,A}(t) == 0 \wedge \mathcal{V}_{r_i\_x-,A}(t) == 1 \rightarrow (y^-, 0)$

**Rule2:**  $C_{r_i} == 'A' \wedge \mathcal{V}_{r_i\_any,B}(t) == 0 \wedge \mathcal{V}_{r_i\_Q3,A}(t) == 0 \wedge \mathcal{V}_{r_i\_Q4,A}(t) == 0 \wedge \mathcal{V}_{r_i\_y-,A}(t) == 0 \wedge \mathcal{V}_{r_i\_x+,A}(t) == 0 \wedge \mathcal{V}_{r_i\_x-,A}(t) == 0 \rightarrow (x^+, 0)$

**Rule3:**  $C_{r_i} == 'A' \wedge \mathcal{V}_{r_i\_any,B}(t) == 0 \wedge \mathcal{V}_{r_i\_Q3,A}(t) == 0 \wedge \mathcal{V}_{r_i\_Q4,A}(t) == 0 \wedge \mathcal{V}_{r_i\_y-,A}(t) == 0 \wedge \mathcal{V}_{r_i\_x+,A}(t) == 0 \wedge \mathcal{V}_{r_i\_x-,A}(t) == 0 \wedge \mathcal{V}_{r_i\_Q1,A}(t) == 0 \rightarrow (0, B)$

---

状況 A のとき、ライトの色が‘ $A$ ’のロボットは、ライトの色‘ $B$ ’のロボットを観測できるならば第四象限に 1 台だけ観測できる。ここで、ロボット  $\alpha$  が存在する  $y$  軸を基準軸、基準軸から 1 だけ小さい  $y$  軸を仮基準軸と呼ぶことにする。ライトの色‘ $B$ ’のロボットを第四象限のみに 1 台だけ観測したライトの色‘ $A$ ’のロボットは、仮基準軸上まで  $x$  軸に沿って移動する。仮基準軸上に存在するロボットは自身よりも  $y$  座標値の小さいライトの色‘ $A$ ’のロボットを観測できないならば、 $x$  軸に沿って基準軸上に移動する。このロボットを基準ロボット  $\beta$  と呼ぶ。ロボット  $\beta$  は同一  $y$  軸のマイナス方向にライトの色‘ $B$ ’のロボット(基準ロボット  $\alpha$ )を観測し、ライトの色を‘ $B$ ’に変更する。ロボット  $\beta$  がライトの色を‘ $B$ ’に変更したとき、状況は状況 B となる。

---

**Algorithm 2** 直線形成フェーズ-状況 A から状況 B

---

時刻  $t$  におけるロボット  $r_i \in \mathcal{R}$  の観測結果を  $\mathcal{V}_{r_i}(t)$  とする。

**Rule4:**

---

文 献

- [1] Ichiro Suzuki, Masafumi Yamashita. Distributed Anonymous Mobile Robots: Formation of Geometric Patterns. SIAM Journal on Computing, Vol. 28, Issue4, pp. 1347-1363, 1999.
- [2] A. Dutta, S. G. Chaudhuri, S. Datta, and K. Mukhopadhyaya. Circle formation by asynchronous fat robots with limited visibility. In Proc. of the 8th International Conference on Distributed Computing and Internet Technology (ICDCIT 2012), p 83-93, 2012.

- [3] P. Flocchini, N. Santoro, G. Viglietta, M. Yamashita. Rendezvous with Constant Memory. Theoretical Computer Science, Vol.621, pp.57-72, 2016.
- [4] Luna,G.A.D., Flocchini,P., Polonini,F., Sanotoro,N., Viglietta,G. :The mutual visibility problem for oblivious robots.In:Proceedings of the 26th Canadian Conference on Computational Geometry,CCCG 2014,Halifax,Nova Scotia,Canada(2014).
- [5] Luna,G.A.D., Flocchini,P., Chaudhuri,S.G., Poloni,F., Santoro,N., Viglietta,G. :Mutual visibility by luminous robots without collisions.Inf.Comput.254,392-418(2017).
- [6] S.Terai., K.Wada., Y.Katayama. :Gathering Problems for Autonomous-Mobile Robots with Lights.arXiv:1811.12068v1[cs.DC].