数理解析研究所講究録
第 1941 巻 2015 年 31-36
2014 Winter LA [C-3]

The Team Assembling Problem for Heterogeneous Mobile Robots

Zhiqiang Liu, Yukiko Yamauchi, Shuji Kijima, Masafumi Yamashita

Kyushu University

1 Introduction

Suppose that there are $2n$ atoms of hydrogen and n of oxygen, without any control form outside, they move adaptively and form n molecules of water, each of which consists of 2 atoms of hydrogen and 1 of oxygen. In the same way, consider a robot system consisting of heterogeneous robots, each robot executes an algorithm and moves adaptively based on the movement of other robots, so that as a whole robots will form some teams, each of which consists of specific types and specific numbers of robots. This process can be called *team assembling* by heterogeneous mobile robots.

The *team assembling problem* by heterogeneous mobile robots is to design a distributed algorithm that is executed on each robot to navigate it so that the robots as a whole eventually form some teams, each of which satisfies a given team specification defined by the number of members for each type.

This paper newly introduces and investigates the *team assembling problem* by a set of *heterogeneous, asynchronous* robots. We denote the heterogeneity by multiple colors of robots in this paper.

We provide a distributed algorithm ϕ that solves the team assembling problem through two phases: it first makes all robots form a circle to reduce the problem into an easier problem of team assembling, then it selects l team leaders to assemble a team at each of the leaders' positions. Moreover, we show a necessary condition that team assembling problem is solvable only if $GCD(a_1, a_2, \cdots, a_k) = 1$, where GCD denotes the greatest common divisor, and $a_i \in N$ $(1 \leq i \leq k)$, denotes the number of robots of color c_i in each team.

Related work The *pattern formation* problem by homogeneous robots has been discussed extensively in the previous research, in which robots must arrange themselves to form a given geometric pattern F from an initial configuration I in a finite number of steps. Based on the *pattern formation* problem, we increase the types of robots, robots of each type form the same geometric pattern respectively. If these geometric patterns overlap (positions of robots) completely, this can be considered as the *team assembling* problem by heterogeneous robots. Moreover, the symmetry breaking is the main difficulty in both of them. The method of breaking the symmetry in the pattern formation problem can be applied in the team assembling problem. Thus, the investigation on the *pattern formation* problem and its related problems is vital.

Suzuki and Yamashita [3] first proposed a computational model of autonomous mobile robots, and they showed that a geometric pattern F is formable by non-oblivious robots from the initial configuration I if and only if the symmetricity of I divides that of F. For oblivious robots, Fujinaga et al. [1] showed embedded pattern formation problem is solvable by asynchronous robots through the optimum matching between the robots and the pattern's points, where they assumed that the pattern's points are visible to all robots. Then, in [2], Fujinaga et al. showed that the pattern formation problem by oblivious robots in the asynchronous model is formable if the symmetricity of I divides that of F, I and F do not contain multiplicities (thus this result also holds for the semi-synchronous model and the fully synchronous model).

The rest of this paper is organized as follows: In section 2, we introduce our model of robots. In section 3, we formally define the problem of team assembling, and show our main result on this problem. In section 4, we present the algorithm and the necessary condition of team assembling. At last, we conclude this paper in section 5.

2 System Model

The system consists of a set of heterogeneous robots in a 2-dimentional Euclidean plane, where each robot is modeled as a colored point. Let $C = \{c_1, c_2, \cdots, c_k\}$ $(k > 1)$ be a set of k colors and $R = \{r_1, r_2, \cdots, r_n\}$ be a set of n robots. We introduce a function c on r_i for all $i = 1, 2, \cdots, n$, such that $c : R \rightarrow C$ $(1 \leq j \leq k, c_j \in C)$. The robots are *heterogeneous* in the sense by (1) the robots can identify the colors of all robots, (2) there is no way to identify robots of the same color (we use r_i just for notation), and (3) the robots of the same color execute the same algorithm.

Let Z_0 be the global $x - y$ coordinate system. Each robot r_i does not know the global coordinate system, however, each robot r_i has its own local x-y coordinate system Z_i, where the origin of the local coordinate system is the current position of r_i, and the unit distance never change. We assume that Z_0 and $Z_i's$ are all right-handed. Hence, the robots have the sense of clockwise and counter-clockwise directions. Let $p_i(t) \in \mathbb{R}^2$ be the location of robot r_i (in Z_0) at time t, where $\mathbb{R}$ is the set of real numbers. Then the multiset $C(t) = \{(p_i(t), c(r_i)) | r_i \in R\}$ is called the configuration at time t. The occurrences of an element is called its multiplicity.

Each robot repeats a *Look-Compute-Move cycle*. In a *Look phase*, the robot obtains the positions and colors of other robots in its local coordinate system, in a *Compute phase*, it computes its next location and track to the next location with a given algorithm, and in a *Move phase*, it moves to the next location along the computed track.

We consider discrete time $0, 1, \ldots$. Then an infinite sequence $\mathcal{E}$: $C(0), C(1), \cdots$ is called an *execution* with an initial configuration $I = C(0)$. We assume that I does not contain multiplicities, i.e., all elements in $C(0)$ are distinct.

With respect to the activation schedule model, three main models have been discussed in the literatures, i.e., the *fully synchronous model*, the *semi-synchronous model* and the *asynchronous model*. In the *fully synchronous model* (FSYNC), all robots synchronously execute a Look-Compute-Move cycle at each discrete time $0, 1, \ldots$. In the *asynchronous model* (ASYNC), robots asynchronously execute each of their Look, Compute, Move phases, which may be interleaved. For instance, a robot may finish the Look phase but another robot may just start its Look phase. The *semi-asynchronous model* (SSYNC) is a particular model between *fully synchronous model* and *asynchronous model*, in which some robots may not start their Look-Compute-Move cycle, but all of those who start the cycle synchronously execute each of its Look, Compute and Move at each discrete time $0, 1, \ldots$. In our paper we consider an *asynchronous* model (ASYNCH).

3 Problem definition and terminology

3.1 Problem definition

We call a k-tuple $A = (a_1, a_2, \cdots, a_k)$ a specification of a team. Given a specification A and an l $(l \in N)$, we say that an algorithm ϕ solves the team assembling problem from any initial configuration $C(0)$ of a set of robots $R = \{r_1, r_2, \cdots, r_n\}$ such that $|\{r_i | c(r_i) = c_j\}| = l \cdot a_j$ for any $j = 1, 2, \cdots, k$, if for any execution $\mathcal{E}$: $C(0), C(1), \cdots$, there is a time instant t_0 such that for any t $(t \geq t_0)$, (1) $C(t) = C(t_0)$ and (2) there exists l distinct points $q_1, q_2, \cdots, q_l$ in $C(t)$, and for each point q_f $(1 \leq f \leq l)$, $|\{r_i | p_i(t) = q_f, c(r_i) = c_h\}| = a_h$ holds for all h $(1 \leq h \leq k)$.

3.2 Terminology and Main result

The lexicographic order. Let a word be a sequence of positive numbers. The *lexicographic order* on two words $X = x_1 x_2 \cdots x_n$ and $Y = y_1 y_2 \cdots y_n$ is the relation defined by $X < Y$ if and only if there is a positive integer t $(t \leq n)$ such that $x_i = y_i$ and $x_{t+1} < y_{t+1}$ holds for every positive integer $i \leq t$.

The greatest common divisor. Let $a_1, a_2, \cdots, a_k$ be k integers, then by $GCD(a_1, a_2, \cdots, a_k)$, we denote the greatest common divisor of $a_1, a_2, \cdots, a_k$.

Main result In this paper, we prove the following theorem.

Theorem 3.1. *For any initial configuration I not containing multiplicities, the team assembling problem is solvable from I by oblivious asynchronous* *robots if and only if GCD* $(a_1, \cdots, a_i, \cdots, a_k) =$ *1.*

4 Team Assembling Algorithm

In this section, we provide our algorithm ϕ to solve the team assembling problem in such a way that it first navigates all robots on a circle (*Circle formation*), then selects l distinguishable and invariant robots as team leaders (Team leader selection), finally makes all non-leader robots gather at the positions where team leaders locate (Team assembling).

4.1 Circle formation

Since the *circle formation problem* has been extensively investigated in the previous research, we just use the existing algorithm ψ in [2] by Fujinaga et al. Note that in our robot system robots have multiple colors, we use the existing method by ignoring the colors.

We would like to explain an outline of algorithm ψ, which consists of four phases. Let I and F denote the initial configuration and terminal configuration, respectively. Phase 1 is to form a regular n-gon H with a radius μ smaller than $l(F)/2$ and the center being $c(I)$, where $l(I)$ denotes the radius of the largest empty circle (its interior does not include a point) in I, and $c(I)$ denotes the center of I. Let $F' = F \backslash H$, Phase 2 and Phase 3 are to roughly solve the formation problem for F', by invoking the modified CWM (clockwise matching). In Phase 4, the robots in H resume their correct positions.

4.2 Team leader selection

In this subsection, we start with a configuration I' where all robots locate on the circumference of the smallest enclosing circle, i.e., the output of circle formation. Firstly, we describe each robot r_i's local view through its own local x-y coordinate, and show that all robots's local views can be ordered globally, irrespective of the coordinate system of each r_i. Using the total order, we provide an algorithm A_p to select l distinguishable team leaders.

Local view. Without loss of generality, we assume that $R = \{r_1, r_2, \cdots, r_n\}$ appear clockwise on circumference. Let $p_1, p_2, \cdots, p_n$ be the locations of these robots, and $c(r_1), c(r_2), \cdots, c(r_n)$ be the colors of these robots. For any two points p_i and p_j, we denote the length of arc $\widehat{p_i p_j}$ from p_i to p_j in the clockwise direction by $\ell(p_i, p_j)$, and let $\ell_i = \ell(p_i, p_{(i+1)})$. Thus, a distance tuple $L_i = (\ell_i, \ell_{(i+1)}, \cdots, \ell_{(i-1)})$ denotes the distance distribution for r_i. Note that, though the unit distance of the coordinate system of each robot is arbitrary, all robots can agree the ordering of all $L_i's$. Moreover, let $Col_i = (c(r_i), c(r_{(i+1)}), \cdots, c(r_{(i-1)}))$. Col_i means the colors distribution for r_i. Thus, for each robot r_i, its local view is $\Phi_i = (L_i, Col_i)$. Roughly, Φ_i means the distribution of colored points around $c(P)$ starting from r_i.

The total order. We apply the lexicographic order to Φ_i, i.e., (1) $\Phi_i < \Phi_{i'}$ holds if and only if $L_i < L_{i'}$, or $L_i = L_{i'}$ and $Col_i < Col_{i'}$; (2) $\Phi_i = \Phi_{i'}$ holds if and only if $L_i = L_{i'}$ and $Col_i = Col_{i'}$, in which we consider color c_j as integer j. As shown in Fig 1, since ℓ_1 is larger than ℓ_i for any $i = 2, 3, \cdots, 15$, we have L_1 is larger than L_i for any $i = 2, 3, \cdots, 15$, furthermore, Φ_1 is the largest globally. The following Lemma 4.1 show that each robot can obtain all robots's local views, which implies that the global order also adapts to each robot r_i.

Lemma 4.1. *Each robot r_i can obtain all robots's local views irrespective of its local coordinate system.*

Team leader selection algorithm A_p. Without loss of generality, we assume that r_1 has the largest local view and $c(r_1) = c_k$ (if $\ell_1 > \ell_i$ for all $i = 2, 3, \cdots, n$, we use $c(r_1) = c_k$ just for the purpose of description, if $\ell_1 = \ell_2 = \cdots = \ell_n$, $c(r_1) = c_k$ holds actually). Algorithm A_p executes in the following sense: starting from a robot with the same local view of r_1 (the largest local view), A_p picks up a robot in every a_k robots from all robots of color c_k, as team leaders. For example, given $A = (1, 4)$ and $l = 3$, in Fig 1, starting from r_1, A_p picks up a robot in every 4 robots from all black (c_2) robots, i.e., r_1, r_6, r_{11}, as team leaders. We consider such l team leader robots distinguishable since the definition of the largest local view.

Note that since there may be multiple robots with the largest local view, A_p may start from

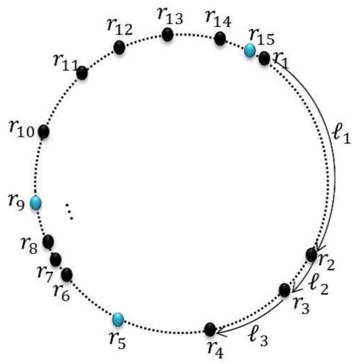

Figure 1: ℓ_1 is larger than ℓ_i for any $i = 2, 3, \cdots, 15$.

other robot other than r_1. Actually, starting from whichever of robots with the largest local view, team leader robots selected by A_p are invariant if $GCD(a_1, a_2, \cdots, a_k) = 1$. The following Lemma 4.2 shows the result.

Lemma 4.2. *A_p selects l invariant team leader robots if $GCD(a_1, a_2, \cdots, a_k) = 1$.*

4.3 Team assembling

In this section, we introduce an *oblivious* algorithm ϕ to solve the team assembling problem in the following procedure: each robot r_i firstly confirms the team leaders by A_p. If r_i is a team leader robot, it stops at position where it locates initially, otherwise r_i moves until it stops at a position where the team leader locates through four phases, whose terminal configurations are F_1, F_2, F_3 and F_4 respectively. For each phase, we show that team leaders selected by A_p never change during or after the movement of non-leader robots.

Configuration F_1 : Let $q_1, q_2, \cdots, q_l$ be locations where team leaders locate, and ξ_i be the middle point of the arc $\widehat{q_i q_{(i+1)}}$ for $i = 1, 2, \cdots, l$, where $q_{(l+1)} = q_1$. F_1 is such a configuration that all non-leader robots locate on the arcs $\widehat{\xi_i q_{(i+1)}}$ clockwise.

Let ϵ be a sufficiently small number, we consider it is less than l_i for any $i = 1, 2, \cdots, n$. Algorithm 1 (the first phase of ϕ) describes the formation of F_1 staring form I'.

Algorithm 1 (for a non-leader robot r_i)

if r_i is active **then**
  r_i moves toward r_{i+1} along the circumference clockwise until $\ell_i = \epsilon$, where i is the smallest integer such that $\ell_i > \epsilon$ holds in each arc $\widehat{q_i q_{(i+1)}}$.
else
  stay still
end if

Lemma 4.3. *Algorithm 1 can form configuration F_1.*

Proof. We give an example to explain the correctness. As shown in Fig 1, since r_1, r_6, r_{11} are team leaders, $i = 2, 7, 12$ are the smallest subscript such that $l_i > \epsilon$ in each leaders interval. Suppose that only non-leader robot r_7 is active, Algorithm 1 orders r_7 to move towards r_8 until $\ell_7 = \epsilon$ in the clockwise direction. Then r_8 moves towards r_9 until $\ell_8 = \epsilon$. Such executing continues until r_7 pass ξ_3. Even though at a time instant, $\ell_6 > \ell_1$ may hold, team leaders selected by A_p never change. □

Configuration F_2 : F_2 is such a configuration that all non-leader robots locate on the arcs $\widehat{\xi_i q_{(i+1)}}$ for all $i = 1, 2, \cdots, l$, where $q_{(l+1)} = q_1$, and all c_k color robots locate on positions $q_1, q_2, \cdots, q_l$ such that $|\{r_i | c(r_i) = c_k, p(r_i) = q_f, 1 \leq f \leq l\}| = a_k$ holds.

Algorithm 2 (the second phase of ϕ) describes the formation of F_2.

Lemma 4.4. *Algorithm 2 can form configuration F_2.*

Proof. Point: since all non-leader robots locate on the second haves of each $\widehat{q_i q_{(i+1)}}$, any two non-leader robots will never leave an empty arc greater than $\widehat{q_i q_{(i+1)}}/2$. □

Configuration F_3 : F_3 is such a configuration that all robots of color c_k gather at positions where leaders locate with the requirement of $|\{r_i | c(r_i) = c_k, p(r_i) = q_f, 1 \leq f \leq l\}| = a_k$, and in each $\widehat{q_i q_{(i+1)}}$, robots of other colors meet the team specification in number, i.e., $|\{r_i | c(r_i) = c_j, r_i \in \widehat{q_f q_{(f+1)}}, 1 \leq f \leq l\}| = a_j$ for all $j = 1, 2, \cdots, (k-1)$ holds, where $q_{(l+1)} = q_1$.

Algorithm 2 (for a non-leader robot r_i)

if r_i observes that F_1 is formed **then**
  if r_i is a robot of color c_i for any $i = 1, 2, \cdots, (k-1)$ **then**
    it stops moving
  end if
  if r_i is a robot of color c_k **then**
    it moves towards and gathers at position where team leader locate clockwise, during which if there are robots of the same color c_k on its way, r_i stops moving, otherwise, r_i moves to its destination.
  end if
else
  it executes Algorithm 1.
end if

Algorithm 3 (the third phase of ϕ) describes the formation of F_3.

Algorithm 3 (for a non-leader robot r_i)

if r_i observes that F_2 is formed **then**
  if r_i is a robot of color c_k **then**
    it stops moving.
  end if
  if r_i is a robot of color c_j for any $j = 1, 2, \cdots, (k-1)$ **then**
    it executes the following
    if $|\{r_h | r_h \in \widehat{p_i q_f}, c(r_h) = c(r_i) = c_j, 1 \leq j \leq (k-1)\}| > a_j$, where q_f is the position of the nearest leader clockwise. **then**
      r_i moves counter clockwise until it passes $q_{(f-1)}$.
    end if
  end if
else
  it executes Algorithm 2.
end if

Lemma 4.5. *Algorithm 3 can form configuration F_3.*

Proof. Point: since robots are *heterogeneous*, they can identify the color (i.e., c_k) of team leaders, and all c_k color robots locate at locations where leaders locate, which results in no leader candidates left. The team leader robots can be identified by color simply. □

Configuration F_4: F_4 is such a configuration that for each point q_f $(1 \leq f \leq l)$, $|\{r_i | p(r_i) = q_f, c(r_i) = c_h\}| = a_h$ for all $h(1 \leq h \leq k)$. That is, F_4 is the terminal configuration in our paper.

Algorithm 4 (the fourth phase of ϕ) describes the formation of F_4.

Algorithm 4 (for a non-leader robot r_i)

if r_i observes that F_3 is formed **then**
  if r_i is a c_k color robot **then**
    it stops moving.
  end if
  if r_i is a c_j color robot for any $j = 1, 2, \cdots, (k-1)$ **then**
    it moves towards the position where team leader locate clockwise, until it reaches it.
  end if
else
  it executes Algorithm 3.
end if

Lemma 4.6. *Algorithm 4 can form configuration F_4.*

4.4 Necessary condition

Lemma 4.7. *There exists an initial configuration, form which robots can not assemble l teams if $GCD(a_1, a_2, \cdots, a_k) > 1$ by any oblivious algorithm.*

Proof. We give an example to explain the correctness. Consider the case with $A = (2, 2)$ and $l = 2$, as shown in Fig 2. Since robots r_1, r_3, r_5, r_7 have the same color and they locate on the vertex of a regular 4-gon, there is no way to arrange 2 robots from them to be in one team. This factor also holds for the robots r_2, r_4, r_6, r_8 of other color. □

5 Conclusion

In this paper, we showed an *oblivious* algorithm ϕ to solve the team assembling problem by *heterogeneous, mobile,* and *asynchronous* robots with the condition $GCD(a_1, a_2, \cdots, a_k) = 1$, from *Circle formation, Team leader selection* and *Team assembling.*

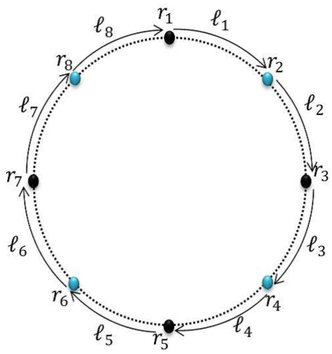

Figure 2: $\ell_1 = \ell_2 = \cdots = \ell_8$.

References

[1] N. Fujinaga, H. Ono, S. Kijima and M. Yamashita: Pattern Formation through Optimum Matching by Oblivious CORDA robots. OPODIS 2010. LNCS, vol.6490, pp. 1-15. Springer, Heidelberg (2010).

[2] N. Fujinaga, Y. Yamauchi, S. Kijima and M. Yamashita: Asynchronous pattern formation by anonymous oblivious mobile robots. In the 26_{th} International Symposium on Distributed Computing, pp. 84 (2012).

[3] I. Suzuki and M. Yamashita: Distributed anonymous mobile robots: Formation of geometric patterns. SIAM Journal on Computing, 28, pp. 1347 - 1363 (1999).

[4] M. Yamashita and I. Suzuki: Characterizing geometric patterns formable by oblivious anonymous mobile robots, Theoretical Computer Science, 411, pp. 2433-2453 (2010).