MOBLOT: Molecular Oblivious Robots

Serafino Cicerone
Dept. of Engineering and Information Science and Maths
University of L'Aquila, L'Aquila, Italy
serafino.cicerone@univaq.it

Alessia Di Fonso
Dept. of Engineering and Information Science and Maths
University of L'Aquila, L'Aquila, Italy
alessia.difonso@graduate.univaq.it

Gabriele Di Stefano
Dept. of Engineering and Information Science and Maths
University of L'Aquila, L'Aquila, Italy
gabriele.distefano@univaq.it

Alfredo Navarra
Dept. of Maths and Computer Science
University of Perugia, Perugia, Italy
alfredo.navarra@unipg.it

ABSTRACT

In swarm robotics, research mainly follows a theoretical approach that considers robot systems in the abstract, where the complexity and capabilities of the underlying model are often reduced to their minimum. One general and well-investigated model is OBLOT, where the robots are silent, anonymous, and *oblivious*.

In this work, we introduce MOBLOT, a model that extends OBLOT to address a larger spectrum of cases. MOBLOT stands for *molecular oblivious robots*: like atoms combine themselves to form molecules, in MOBLOT simple robots can move to form more complex computational units, having an extent and different capabilities with respect to robots; like molecules combine themselves to form the matter, in MOBLOT the complex structures can exploit their own capabilities to arrange themselves to form any shape defining an acceptable final structure. In MOBLOT, we formally define the *Matter Formation* (MF) problem and, as a preliminary general result, we provide a necessary condition for its solvability which relies on *symmetricity*. Informally, the symmetricity of a configuration measures the amount of symmetries of the robots' disposal. We actually show how dealing with molecules can resolve in some cases the symmetry breaking issue where OBLOT cannot. Finally, we provide a case study for MOBLOT, that is, a representative MF problem along with a resolution distributed algorithm.

KEYWORDS

Swarm Robotics; Self-Organizing Systems; Oblivious Robots; Pattern Formation

ACM Reference Format:
Serafino Cicerone, Alessia Di Fonso, Gabriele Di Stefano, and Alfredo Navarra. 2021. MOBLOT: Molecular Oblivious Robots. In *Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), Online, May 3–7, 2021*, IFAAMAS, 9 pages.

1 INTRODUCTION

Robotics is an interdisciplinary research area at the interface of computer science and engineering. Robotics involves design, construction, operation, and use of robots. Among many, two well-established research areas in robotics concerns *modular robotics* (see, eg [2, 5, 22, 29]) and *swarm robotics* (see, eg [3, 27]).

Modular robots refer to robotic systems composed of interconnected individual (electro-mechanical) modules that can rearrange in order to best adapt to their task-environment or recover from failures (eg, see [30]). Their promise is to realize robotic systems that are more versatile, affordable, and robust than their conventional counterparts, at the cost of a probable reduced efficacy for specific tasks.

Swarm robotics differs from modular robotics by the fact that the individual robots in the system don't have to remain connected to each other at all times, but they are usually mobile units with full autonomy (eg, Kilobot [26]). In this area, it is supposed that a desired collective behavior emerges from some form of interaction between the robots. Researchers in the field of swarm robotics mainly follow a theoretical approach that considers robot systems in the abstract, where the complexity and capabilities of the underlying model are often reduced to their minimum. As an example, consider the well-known Amoebot model [17, 18], and the recent models Silbot[15, 16], and Pairbot [23]. In general, these models facilitate rigorous algorithmic analysis, thus providing new theoretical insights to expand the practical possibilities of the studied systems.

One general and well-investigated theoretical model for swarm robotics is OBLOT (see, eg [20]), where the robots are computational entities that can move in some environment. The OBLOT model covers a large spectrum of settings, each defined by specific choices among a range of possibilities, wrt a fundamental component - time synchronization - as well as other important elements, like memory, orientation and mobility. Such settings are often maintained at minimum: robots are assumed to have no memory about past activities, to be indistinguishable, without ids, without any centralized control, they all execute the same algorithm without any means of direct communication and each one operating with its own local coordinate system. When active, a robot operates in three phases: it takes a snapshot of the current global configuration in terms of robots' positions according to its own coordinate system; successively, it decides whether to move toward a specific target or not; in the positive case it moves. In the OBLOT model, one of the most studied problem is certainly the *Gathering* [8, 10, 12, 13], or the more general *Pattern Formation* (PF) [6, 7, 9, 11, 21, 25, 28]: given a team of robots and a geometric pattern in terms of points in the plain wrt an ideal coordinate system, the goal is to design a distributed algorithm so that eventually all robots together form the pattern, if possible.

In this work, we introduce MOBLOT, a new theoretical model in the context of swarm robotics that extends OBLOT to address a

Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), U. Endriss, A. Nowé, F. Dignum, A. Lomuscio (eds.), May 3–7, 2021, Online.

larger spectrum of problems. MOBLOT stands for *Molecular OBLivious robOTs*, since the inspiration comes from nature: like atoms combine themselves to form molecules, in MOBLOT simple robots can move to form more complex computational units (called *molecules* also in the model), having an extent and different capabilities wrt robots; like in nature molecules combine themselves to form the matter, the MOBLOT version of molecules can exploit their own capabilities to arrange themselves to form any shape defined according to some compositional properties. This is what we later call the *Matter Formation* (MF) problem which is well-related to PF. Once MF is solved, in case of other inputs/stimuli/requirements, the molecules can rearrange (self-reconfigure) their positions to modify the shape or collectively move. This highlights how MOBLOT is capable of modeling problems also in the context of modular robotics.

As a general result, we provide a necessary condition for the solvability of MF which relies on *symmetricity*. Informally, the symmetricity of a configuration measures the amount of symmetries of the robots' disposal. We actually show how dealing with molecules can resolve in some cases the symmetry breaking issue where OBLOT cannot. Finally, we provide a case study, that is, a representative MF problem along with a resolution distributed algorithm.

Note that, robots/molecules we deal with can be thought as oblivious agents, i.e., without memory of past events. Hence, they are in general much weaker than standard agents (see, eg [19]), without any means of learning. Their behavior is in fact provided via a deterministic algorithm and it cannot change over time.

2 MOLECULAR OBLIVIOUS ROBOTS

In this section, we first recall from [20] the OBLOT model and then present the MOBLOT extension.

An OBLOT system is composed by a set $R = \{r_1, r_2, \ldots, r_n\}$ of n computational entities, called robots, that live and operate in a connected spatial universe $\mathcal{U} \subseteq \mathbb{R}^d$, $d \geq 1$, in which they can move. Robots are viewed as points in $\mathbb{R}^d$ (they are **dimensionless**), and more than one robot can occupy the same location at the same time; when this occurs, we say that there is a **multiplicity**.

The robots have the following basic properties: they are **identical** (they are indistinguishable by their external appearance), **anonymous** (they do not have distinct identities that can be used during the computation), **autonomous** (they operate without a central control or external supervision), **homogeneous** (they all have and execute the same protocol, or algorithm), **silent** (they have no means of direct communication of information to other robots), and **disoriented** (each robot has its own local coordinate system - LCS). A robot is capable of observing $\mathcal{U}$ hence determining the positions (expressed in its LCS) of all the robots. It follows that the only means of interaction between robots are observations and movements, that is, communication is *stigmergic*. The behavior of a robot follows four sequential phases:

- `Wait`. The robot is idle. A robot cannot stay indefinitely idle.
- `Look`. The robot observes $\mathcal{U}$ obtaining a snapshot of the positions of all other robots expressed in its own LCS.
- `Compute`. The robot performs a local computation according to a deterministic algorithm $\mathcal{A}$ (we also say that the robot executes $\mathcal{A}$). The algorithm is the same for all robots, and the result of the `Compute` phase is a destination point along with the trajectory to be followed.
- `Move`. The robot moves towards the computed destination; if the destination is the current location, the robot stays still, performing a nil movement.

Such phases form a **computational cycle** of a robot. Robots are **oblivious**: robots have no memory of past actions, and the computation is based only on what determined in their current cycle. In particular, from the snapshot acquired during the `Look` phase, a robot can elaborate what later is called its **view**. This is a data structure containing all the information acquired by a robot during its `Look` phase. Since each robot refers to its own LCS, the view cannot exploit absolute measurements but it is based on relative angles and positions of robots. Hence, if symmetries occur, then symmetric robots have the same view if they are both activated while the same configuration occurs. In turn, (i) the algorithm cannot distinguish between symmetric robots (even when placed in different positions), and (ii) symmetric robots that have perceived the same view perform the same movements, eventually.

2.1 Varying the components of the system

The OBLOT model is based on the fundamental properties recalled so far. Anyway, it can address a larger spectrum of situations by varying some additional components.

Time scheduler. Among the most important features that can greatly vary the computational power of robots there is the *time scheduler*. Three schedulers are in general used:

- In the *Semi-synchronous* (SSync) scheduler, the activations of the robots is logically divided into global rounds; in each round, one or more robots are activated and obtain the same snapshot; based on that snapshot, they compute and perform their move, ending their cycle by the next round. The choice of which robots are activated in a given round is assumed to be made by the time scheduler.
- The *Fully-synchronous* (FSync) scheduler is an extreme case of time scheduler: all the robots are activated in every round;
- *Asynchronous* (Async): The robots are activated independently, and the duration of each phase is finite but unpredictable. In other words, robots do not have a common notion of time. As a result, computations can be made based on totally obsolete observations, taken arbitrarily far in the past. Moreover, they can be seen while moving, and computations can be made based on obsolete information about positions.

In SSync and Async, the time scheduler is **fair**: for every robot r and time t, there exists a time $t' \geq t$ at which r is activated; that is, every robot is activated infinitely often.

Orientation. Concerning the orientation, we have already recalled that, in general, robots are assumed to be disoriented: each of them has its own LCS and its unit of measure. It is possible to customize the system by assuming that all robots agree on the direction and orientation of k axes ($1 \leq k \leq d$). Moreover, robots may have *chirality*, that is they agree on a cyclic orientation (eg, clockwise) of the plane.

Mobility. The actual movement of a mobile robot is controlled by an external *mobility scheduler*. The scheduler decides how fast the robot moves toward its destination point, and it may even interrupt its movement before the destination point is reached. In Async, two variants can be defined: *rigid* (or unlimited) mobility, where all robots always reach their destinations when performing Move; *non-rigid*, where the distance traveled within a move is neither infinite nor infinitesimally small. More precisely, the mobility scheduler has also the power to stop a moving robot before it reaches its destination, but there exists an unknown constant $\delta > 0$ such that if the destination point is closer than δ, the robot will reach it, otherwise the robot will be closer to it of at least δ.

Extent. In the standard model, robots are viewed as points; i.e., they are dimensionless. This property can be varied by assuming robots with a physical dimension, that is, entities with an extent. These robots are called *solid* (or *fat* as in [1, 4, 14]) and are viewed as opaque circular disks of a fixed diameter (hence they are assumed to have a common unit distance).

Adversary. The mobility scheduler as well as the time scheduler are both managed by an ideal *adversary*. In fact, such schedulers are completely out of the control of the robots.

2.2 The Pattern Formation problem

Regardless of the adversary, the activations of the robots determine specific ordered time instants. Let $R(t)$ be the configuration observed by some robots at time t during their Look phase, and let $\{t_i : i = 0, 1, \dots\}$, with $t_i < t_{i+1}$, be the set of all time instances at which at least one robot takes the snapshot $R(t_i)$. Since the information relevant for the computing phase of each robot is the order in which the different snapshots occur and not the exact time in which each snapshot is taken, then, wlog we can assume $t_i = i$ for all $i = 0, 1, \dots$. It follows that an *execution* of an algorithm $\mathcal{A}$ from an initial configuration R is a sequence of configurations $\mathbb{E} : R(0), R(1), \dots$, where $R(0) = R$ and $R(t+1)$ is obtained from $R(t)$ by moving some robots according to the result of their Compute phase as implemented by $\mathcal{A}$. Moreover, given an algorithm $\mathcal{A}$ in Async or SSync, there exist many executions from $R(0)$ depending on the activation of the robots, controlled by the adversary.

In OBLOT systems, one of the most studied problem is certainly the Pattern Formation (PF). Given a team of robots R and a geometric pattern F in terms of (a multiset of) points in the universe $\mathcal{U}$ wrt an ideal coordinate system, the goal is to design a distributed algorithm $\mathcal{A}$ that works for each robot to guide it so that eventually all robots together form the pattern, if possible. As the global coordinate system is usually unknown to the robots, a pattern is declared formed as soon as robots are disposed *similarly* to the input pattern, that is regardless of translations, rotations, reflections, uniform scalings. Formally, $\mathcal{A}$ *forms* the pattern F from a configuration R if for each execution $\mathbb{E} : R = R(0), R(1), R(2), \dots$, there exists a time instant $t' > 0$ such that $R(t')$ is similar to F and $R(t) = R(t')$ for each time $t \geq t'$.

2.3 The MOBLOT model

It is very common to find self-organizing structure in the physical world. For instance, atoms (the smallest units of matter) first combine with other atoms to form molecules (special kind of atom compounds) and then the molecules combine with each other to form some kind of matter, like for instance a crystal. We use this *matter formation paradigm* to present the MOBLOT model.

In a MOBLOT system, the smallest units correspond to the robots of the OBLOT model with the only difference that robots might be **heterogeneous** (eg, see [24]). This is modeled by assuming that each robot has assigned a *color* taken from a finite set *Col* used to specify its type. However, robots of the same color are identical and homogeneous. With respect to the above paradigm, each robot can be thought as an atom, and its color specifies the type of atom. For instance, the robot resembling the smallest units to form water could be modeled by *white* and *black* robots corresponding to hydrogen and oxygen atoms, respectively.

In a MOBLOT system, the algorithmic task for robots is to form **molecules**. A molecule μ is specified by a **fixed pattern** defined wrt the same universe $\mathcal{U}$ where the robots move. For instance the water molecule is composed by two *white* and one *black* robots, where the white robots form a 104.5° angle with the *black* robot, and each *black*-*white* pair is at distance about 0.096 *nm*.

The minimal ball enclosing a molecule μ is denoted as $B(\mu)$, and its diameter is denoted as $diam(B(\mu))$. It is possible that in $\mathcal{U}$ there are robots that must form different molecules, hence we denote as $\mathcal{M} = \{\mu_1, \mu_2, \dots, \mu_m\}$ the set containing all kinds of molecules to be formed. In order to make the model fair enough and as much general/weak as possible, we impose some constraints:

*C*1: in any initial configuration *R*, each pair of robots is at distance greater than $D = \max\{diam(B(\mu)) \mid \mu \in \mathcal{M}\}$.

Assume that an algorithm $\mathcal{A}$ is able to move robots from an initial configuration R so that at a given time t some molecules in $\mathcal{M}$ are formed. In the MOBLOT system we assume that each robot r performing the Look phase at time t is able to detect not only all the other robots but also any formed molecule μ. Notice that r perceives both $B(\mu)$ and the robots inside the ball, that is $B(\mu)$ is "transparent". According to the ability of perceiving possible formed molecules, and being the molecules expressed as fixed patterns, robots have common **units of length** (for measuring distances, angles, etc).

A molecule μ is formed as soon as there are robots that form the pattern describing μ, however it must hold:

*C*2: in $B(\mu)$, there are only the robots necessary to form μ suitably placed wrt the pattern defining μ;

*C*3: for each μ' already formed or that could be formed at the same time of μ, then $B(\mu) \cap B(\mu') = \emptyset$;

*C*4: assume a robot r is moving along a trajectory τ toward a target t and there is a position $p \neq t$ along τ such that a molecule μ is formed once r in on p; if μ can be formed (i.e., all the previous constraints are fulfilled), then r is automatically stopped at p and the molecule is formed;

*C*5: as soon as a molecule μ is formed, each robot forming μ is no longer an independent computational unit (i.e., it stops executing its algorithm and acts as a part of the molecule).

Once a molecule is formed, it constitutes a new computational entity with a physical dimension, i.e., it is solid. The basic properties of such new entities can still be modeled as in OBLOT systems (and its variants), with the main exception that a molecule not only can move along any trajectory but it also may rotate wrt its center.Being solid, any other element in $\mathcal{U}$ (robot or molecule) can touch the external surface of $B(\mu)$ – but cannot penetrate inside.

Each type of molecule in $\mathcal{M}$ is provided as input to the algorithm, and the algorithm is responsible to assemble all the molecules so that a more complex structure (i.e., the **matter**) is formed. Also the matter to be formed must be given as input to robot/molecules and it can be defined in a very general way according to some **adjacency properties**. These properties just define as the molecules must be close (or in contact) each other, and accordingly the final form may vary. For instance, when forming the water, two molecules are connected when their external surfaces share a point p and the robots in each molecule closest to p have different colors; moreover, each molecule must be adjacent to another one so that the final structure must be connected. According to this definition, the molecules may form the matter by forming one among many possible patterns (by pattern we mean any configuration obtained by placing in the universe the molecule so that their positions fulfill the adjacency properties). We use symbol $\mathcal{F}$ to denote the set containing all the possible patterns describing the matter.

With respect to the defined MOBLOT model, the Matter Formation (MF) problem (which is well-related to PF) can be defined as follows: Given a team of robots R, a set of molecules $\mathcal{M}$, and a set $\mathcal{F}$ of possible patterns describing the matter to form, the goal is to design a distributed algorithm $\mathcal{A}$ that works for each robot and molecule so that eventually they form any pattern in $\mathcal{F}$, if possible.

2.4 Necessary condition on feasibility

In this section, the spatial universe $\mathcal{U}$ is confined to the Euclidean plane (even though the result can be easily extended to higher dimensions). Let P be a set of *colored* points, $C(P)$ be the smallest circle enclosing all the points in P, and $c(P)$ be the center of $C(P)$. If C is any circle in the Euclidean plane, let $\delta(C)$ denote the radius of C. The set P can be decomposed into a set of concentric circles centered in $c(P)$, each containing at least a point of P. Let $C_1^P, C_2^P, \dots$ be all such circles ordered from the smallest to the greatest; formally, given C_i^P and C_j^P, then $\delta(C_i^P) < \delta(C_j^P)$ iff $i < j$. When clear from the context, the superscript will be omitted. It is well-known that there exists a $q \geq 1$ divisor of $|P|$ such that every C_i can be decomposed into a set of q-gons, where each q-gon is composed by points of the same color. If there is a point at $c(P)$, this point is, by definition, a (degenerated) 1-gon. The largest q for which this holds is called *symmetricity* of P and denoted by $\rho(P)$. The set of points in each $\rho(P)$-gon is called *orbit*. Robots that occupy all the points in an *orbit* are called *equivalent*, since no algorithm can distinguish among them. If a move is applied to a robot the adversary can symmetrically move any other equivalent robot.

In the remainder, by $\rho(R)$, $\rho(\mu)$, and $\rho(F)$ we denote, respectively, the symmetricity of a configuration of robots, of the set of robots forming a molecule μ, and of a set of molecules forming any pattern F of a desired matter to be formed.

Theorem 2.1. *Let R be an initial configuration, $\mathcal{M}$ be the set of molecules, and $\mathcal{F}$ be the set of all possible patterns defining the matter. If there exists an algorithm $\mathcal{A}$ able to form the matter, i.e., a pattern $F \in \mathcal{F}$, then (1) $\rho(R)$ divides $\rho(F)$, or (2) there exists $\mu \in \mathcal{M}$ such that $\rho(R)$ divides $\rho(\mu)$.*

Proof. If $\mathcal{A}$ is able to form F without moving the molecules (i.e., matter is formed as soon as molecules are formed), then by [28] we get that property (1) holds. In what follows we assume that $\mathcal{A}$ must move some molecule to form F. We assume $\rho(R) > 1$, otherwise both properties (1) and (2) are trivially verified. Notice that for each possible execution $\mathbb{E} : R = R(0), R(1), \dots$ of $\mathcal{A}$, the adversary may force $\rho(R(0))$ equivalent robots to move synchronously. Let $R(t)$, $t > 0$ be the first configuration containing molecules.

If $R(t)$ contains more than one molecule, according to the synchronous moves and to the symmetricity of R, then (i) in $R(t)$ there are $\rho(R(0))$ molecules, (ii) the molecules in $R(t)$ are all equal, and (iii) $\rho(R(t)) = \rho(R(0))$. Then, from $R(t)$ on, each move planned by $\mathcal{A}$ may be forced by the adversary to maintain at least the same symmetricity $\rho(R(0))$ until F is formed. Then $\rho(R(0))$ divides $\rho(F)$ and property (1) holds.

If $R(t)$ contains just one molecule μ, then it must be formed around the center of the configuration. Even in this case, the adversary forces $\rho(R(0))$ equivalent robots to move synchronously, and then $\rho(\mu)$ must be a multiple of $\rho(R(0))$, i.e. property (2) holds. □

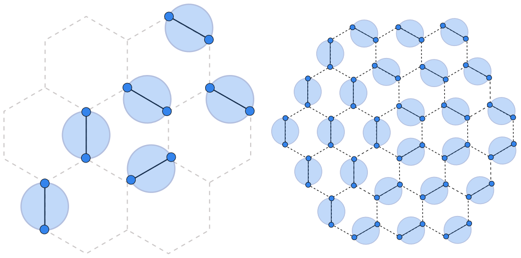

Figure 1: (left) matter composed by 6 molecules; (right) three full levels of the matter. The hexagonal grid emphasizes the relative positions of the molecules composing the matter.

3 CASE STUDY

In this section, we introduce a problem as case study of the MOBLOT model in order to appreciate its facets. In particular, this provides a confirmation that the MOBLOT model is more powerful than OBLOT. First, we introduce some basic notation, necessary for the definition of the problem. After defined the specific problem, we provide a resolution strategy on the basis of the approach defined in [9]. Third, we provide an extensive example in order to better explain how the approach works.

3.1 Problem definition and basic notation

In our MF problem, we assume Async robots that are homogeneous (they all have the same color), and are endowed with chirality, that is a common handedness. We remind that robots also share a common unit of measure and are aware of the quantity D necessary to form molecules. A molecule is defined by two robots at a fixed distance D. Also molecules are assumed to be Async, homogeneous and endowed with chirality. The matter is composed by suitably disposing the molecules as they were on some edges of a hexagonal grid of side D, see Fig. 1. In particular, consider one hexagon of the grid as the core of the matter where three non-adjacent edges of the hexagon correspond to three places where the molecules should lie. Then, the six hexagons surrounding the core represent the place where the second level of the matter would be formed. That is, the non-adjacent edges of the second level, not shared with the first level and parallel to those where the first three molecules are posed, correspond to the second level of edges where molecules can be moved to form the matter, as soon as the first level is full. The i-th level will be formed by the non-adjacent edges of the hexagons surrounding the $(i-1)$-th level, not shared with the $(i-1)$-th level and parallel to those where the molecules of the $(i-1)$-th level are posed. Fig. 1.right shows three complete levels of the matter. Actually, as in Fig. 1.left, the last level of the matter can be not fully occupied. It follows that $\mathcal{F}$ contains all the patterns of molecules that satisfy the above definition of matter.

An initial configuration consists of a set R of robots, with $|R| = 2m$, $m > 3$, and each robot occupying a different point – by the MOBLOT model, it also holds that the distance between any two robots is greater than D.

The goal is to reach a final configuration where m molecules are disposed as an element $F \in \mathcal{F}$. Since $\rho(\mu) = 2$ and $\rho(F) = 1$ or $\rho(F) = 3$, we can assume that $1 \leq \rho(R) \leq 3$, otherwise, by Theorem 2.1, F cannot be formed. We remind that a molecule can be formed if the ball including the two corresponding robots is empty and not overlapping with any other possible molecule. Also, two molecules cannot overlap ever but on a single point.

We denote by Mol the set of molecules formed in a configuration and by $Mat \subseteq Mol$ the subset of molecules forming the matter.

3.2 Description of the algorithm

Informally, in our strategy, first the algorithm ensures to make enough space among the robots before starting forming the matter. Ideally, the matter will be formed around $c(R)$. Such space is realized by moving almost all robots onto a circle C^* having a sufficiently large radius. The center of such a circle is determined by one or three molecules preliminarily formed to this end. Such molecules will constitute the initial core of the matter. However, before letting the molecules suitably disposing to initialize the matter, other molecules are formed on C^* (this is required to assure the correct execution of the algorithm). Once the matter is initialized by means of the internal molecules, those on C^* are moved to join the matter formed so far. The matter formation then proceeds similarly by first forming molecules on C^* and then making them moving to form the matter, until all robots belong to molecules that form the matter.

The strategy has been realized by means of nine tasks (we use a decompositional approach so that each sub-problem is simple enough to be thought as a "task" that can be realized by (a subset of) robots or molecules is allowed to move. The selection of the robots / molecules is based on some predicates computed on the configuration perceived during the Look phase, and it is assumed to resolve possible ties by means of the minimum view if not explicitly specified. Given the perceived configuration, the predicate P_i that results to be true reveals to robots that the corresponding task T_i is the one to be performed. This approach requires that the designed predicates P_i, $1 \leq i \leq 9$ must guarantee some properties:

- Prop_1: *each P_i must be computable on the configuration perceived in each* Look *phase;*
- Prop_2: *$P_i \wedge P_j =$* false, *for each $i \neq j$;*
- Prop_3: *for each possible perceived configuration there must exists a predicate P_i evaluated as true.*

var	*definition*
FarC	All robots (excluding molecules) are on C^*.
iM3′	The distance from $c(R)$ identifies exactly three robots r_1, r_2, r_3 AND let $R' = R \setminus \{r1, r2, r3\}$, $\rho(R') = 3x$, $x > 0$ AND At least one robot among $r1$, $r2$, $r3$ is not part of a molecule AND Let $C_{R'}$ be the smallest circle centered in $c(R)$ and containing robots of R'. The radii passing through r_1, r_2, r_3, resp., and rotating clockwise, meet three robots r'_1, r'_2, r'_3 on $C_{R'}$ at 120^o each other AND r_1, r_2, r_3 are on the same circle OR their projections on $C_{R'}$ coincide with r'_1, r'_2, r'_3, resp.
iM3″	C^R_1 contains more than three robots AND There exist r_1, r_2, r_3 on C_R s.t.: their distance to the next (clockwise) robots is minimum and their rotation toward the next (clockwise) robots generates a configuration R' with $\rho(R') = 3$ AND At least one robot among $r1$, $r2$, $r3$ is not part of a molecule.
cM	$\neg$iM3′ AND $\neg$iM3″ AND $\neg$FarC AND $\lvert Mol \rvert = 1$ OR ($\lvert Mol \rvert = 3$ AND $\rho(Mol) = 3$).
nM1	$\lvert Mat \rvert = 0$ AND FarC AND Let Mol' be the set of molecules inside C^*: $\lvert Mol' \rvert = 1$ OR $\lvert Mol' \rvert = 3$ AND The number of molecules on C^* is less than $\rho(R^+)$.
nM2	$\lvert Mat \rvert = 0$ AND FarC AND $2 \leq \lvert Mol \rvert \leq 3$: 1 or 2 molecules are on C^* and 1 internal AND The number of molecules on C^* no less than $\rho(R)$.
nM3	$0 \leq \lvert Mat \rvert < 3$ AND FarC AND $\lvert Mol \rvert = 6$: 3 molecules are on C^* and 3 internal.
M1	$\lvert Mat \rvert > 0$ AND $\lvert Mol \setminus Mat \rvert < 3$ AND FarC.
M2	$\lvert Mat \rvert > 0$ AND FarC AND ($\lvert Mol \setminus Mat \rvert = 3$ OR ($\lvert Mol \setminus Mat \rvert > 0$ AND the matter does not admit a rotation) OR ($\lvert Mol \setminus Mat \rvert = 1$ AND the matter admits a rotation AND $\rho(R \setminus \{r_1, r_2, r_3\}) = 1$)).
Mf	Matter formed.

Table 1: Boolean variables used to define the preconditions.

If we guarantee that all these properties hold, then the algorithm can be used in the Compute phase as follows: – *if a robot / molecule executing the algorithm detects that predicate P_i holds, then it simply performs move m_i associated with task T_i.*

If we denote by $\mathtt{pre}_i$ the precondition that characterizes the task T_i (see Tables 1 and 2), the corresponding predicate P_i, for each $1 \leq i \leq 9$, can be defined as follow:

$$P_i = \mathtt{pre}_i \wedge \neg(\mathtt{pre}_{i+1} \vee \mathtt{pre}_{i+2} \vee \ldots \vee \mathtt{pre}_8) \qquad (1)$$

Then, each predicate P_i fulfills Property Prop_2. We will show in Section 3.3, by means of a running example, how Properties Prop_1 and Prop_3 too are satisfied by our algorithm.

We are now ready to define the nine tasks that compose our algorithm, exploiting the Boolean variables provided in Table 1. The algorithm is then formalized in Table 2 that refers to Table 3 for the corresponding moves. An example of the execution of the algorithm will be given in Section 3.3. For what follows we encourage the reader to always have an eye on the referred tables.

Task T_1 and T_2. These tasks are mainly due to handle initial configurations where no molecules are formed yet (sub-problem Formation of the Initial Molecules (*FIM*)). According to the symmetricity of the configuration, either 1 or 3 molecules should be formed in order to identify the center of C^*. Task T_1 is applied when the symmetricity

problem	*sub-problem*		*task*	*precondition*	*move*
MF	*FIM*	FIM_1	T_1	*true*	m_1
		FIM_2	T_2	iM3′ $\vee$ iM3″	m_2
	MRA		T_3	cM	m_3
	FM1		T_4	nM1	m_4
	IM	IM_1	T_5	nM2	m_5
		IM_2	T_6	nM3	m_6
	FM2		T_7	M1	m_7
	MM		T_8	M2	m_8
	MD		T_9	Mf	*nil*

Table 2: Algorithm for *MF*.

is either 1 or 2 (sub-problem FIM_1), whereas T_2 is applied when the symmetricity is 3 (sub-problem FIM_1).

In case of T_1 (see, eg Fig. 2.left), let C_m be the circle of radius $D/2$ centered in $c(R)$. Let r_1 be the robot closest to $c(R)$. Let r_2 be the second robot closest to $c(R)$ (excluding r_1) and, in case of ties, the closest to the line passing through r_1 and $c(R)$, on the other side wrt r_1. If r_1 is outside C_m, it moves radially toward C_m, otherwise r_2 moves radially toward C_m. As soon as r_1 and r_2 are at distance D, they form a molecule and the task is over. Note that, when the symmetricity of the initial configuration is 2, then r_1 and r_2 must be antipodal (as in Fig. 2.left). Hence, they form just one molecule (see, eg Fig. 2.right).

In case of T_2, two different subcases are considered, both leading to the formation of 3 symmetric molecules. The three molecules are formed by means of the most internal robots. The two subcases are based on the preconditions iM3′ and iM3″, see Table 1.

If iM3′ holds, then the distance from $c(R)$ identifies exactly three robots r_1, r_2, r_3. In this case, they first rotate clockwise along C^R_1 until they are all aligned with r'_1, r'_2 and r'_3 lying on $C_{R'}$. If they reach such an alignment without creating molecules, then they move radially toward r'_1, r'_2 and r'_3, resp., until forming three molecules.

If iM3″ holds, then there are more than three robots on C^R_1, including r_1, r_2 and r_3. In this case they rotate clockwise along the circle, until three molecules are formed.

Task T_3. This task is used to move all robots (not forming molecules) on C^* (sub-problem Move Robots Away (*MRA*)), as shown in Fig. 2.right. The only molecule formed in T_1 or the three molecules formed in T_2 are used in this task to detect a center from which C^* is identified. In case only one molecule is formed, then C^* is the circle, including all the robots, of minimum radius not smaller than $2mD$ and multiple of $2m$, admitting an annulus A delimited by C^* and a circle of radius $r(C^*) - 3D$ where at most one robot resides. In case three molecules are formed and are included in a circle of radius x, then C^* is the circle, including all the robots, of minimum radius not smaller than $2mD + x$ and multiple of $2m$, admitting an annulus A delimited by C^* and a circle of radius $r(C^*) - 3D$ where at most three robots reside, one for each sector wrt $\rho(Mol)$.

Task T_3 is characterized by precondition cM. In particular, the predicate ensures that one or three molecules have been formed, and in case of one, by $\neg$iM3′ AND $\neg$iM3″ no further molecules must be formed. In order to move the robots on C^* we need to define a suitable set T of target points. If $\rho(Mol) = 1$ let L be the line passing through $c(C^*)$ and orthogonal to the segment between the two robots forming the molecule. Let $P = \{p_1, p_2\}$ with p_1 and p_2 being the intersections of L with C^*; If $\rho(Mol) = 3$ instead, let L_1, L_2 and L_3 be the radii of C^* passing through the center of each molecule, then $P = \{p_1, p_2, p_3\}$ with p_1, p_2 and p_3 being the intersections of L_1, L_2 and L_3, resp., with C^*. The set T is defined by all the points at a distance multiple of $\pi r(C^*)/m$ from points in P in the clockwise direction on C^*. Being C^* of radius multiple of $2m$, the points of T are $2m$, including those in P, and equally distributed on C^*. Robots are moved on C^* so as to not create undesired molecules. For each sector, and in a coordinated 3-steps way, the robot furthest from $c(C^*)$ is first moved radially until distance $1.5D$ from C^* (that is in the exact middle of A), then it rotates clockwise until being on the radius of C^* passing through the first unoccupied target, and finally moves radially to the target. Note that there might be at most three robots moving concurrently. The use of A is to be sure that the moving robots do not create molecules accidentally while moving. In fact the width of A is $3D$ and robots move in the middle of A, that is at distance at least $1.5D$ from any other robot.

Task T_4. This task is devoted to the formation of molecules on C^* (sub-problem Forming Molecules 1 (*FM1*)), as shown in Fig. 3.left. The task is needed to assure that T_3 is finished and that the matter can be formed (by means of tasks T_5 or T_6). Hence T_4 can be thought as an auxiliary task exploited to guarantee the evolution of the system from T_3 to T_5 or T_6. In fact, the formation of the matter without creating the molecules handled by T_4 may result in a modification of the definition of C^*.

T_4 is characterized by precondition nM1. If $|Mol'| = 1$, let $X = \{r_1, r_2\}$ be the first two robots that are met from p_1 and p_2, respectively, in the clockwise direction. If $|Mol'| = 3$, let $X = \{r_1, r_2, r_3\}$ be the first three robots that are met from p_1, p_2, and p_3, respectively, in the clockwise direction. Let $R^+ = R \setminus X$. If $|X| = 3$ or ($|X| = 2$ and $\rho(R^+) = 2$), then all robots in X rotate clockwise; otherwise among r_1 and r_2 the farthest from L rotates clockwise.

In doing so, one, two or three molecules are formed on C^* according to the possible initial symmetry deduced from $\rho(R^+)$. Such molecules along with the positioning of the other robots on C^* allow the internal molecules to move to create the core of the matter made by either one or three molecules.

Task T_5 and T_6. These tasks are due to the movement of the first molecules that start the composition of the matter (sub-problem Initialization of the Matter (*IM*)). According whether there are one (as in Fig. 3.right) or three molecules inside C^*, Task T_5 or Task T_6, resp., is executed. The center of C^* is identified by means of the molecules along with all other robots on C^*. Internal molecules can freely move without changing the identification of C^*.

Task T_5 (sub-problem (IM_1)) is characterized by precondition nM2. The unique internal molecule radially moves along L until reaching a position consistent with the center of a molecule forming the matter wrt $c(C^*)$. Note that if the initial configuration was admitting symmetricity 2, after T_5 the configuration becomes asymmetric.

Task T_6 (sub-problem (IM_2)) is characterized by precondition nM3. The three internal molecules first rotate clockwise wrt to their center until the radii of C^* passing through their centers become orthogonal to the segments joining the two robots forming

move	*definition*
m_1	Let C_m be the circle of radius $D/2$ centered in c_m that coincides with $c(R)$. Let r_1 be the robot closest to c_m (of minimum view in case of ties) and r_2 be the robot closest to c_m (excluding r_1) and the closest to the line passing through r_1 and c_m, on the other side wrt r_1 in case of ties. If r_1 is outside C_m, it moves radially on C_m, otherwise r_2 moves radially toward C_m.
m_2	If `iM3′` holds: r_1, r_2 and r_3 rotate until they are all aligned with r'_1, r'_2 and r'_3. If they reach such an alignment without creating molecules, then they move radially toward r'_1, r'_2 and r'_3, resp.; if `iM3″` holds, then r_1, r_2 and r_3 rotate clockwise.
m_3	For each sector, and in a coordinated 3-steps way, the robot furthest from $c(R)$ is first moved radially until distance $\frac{3}{2}D$ from C^* (that is in the exact middle of A), then it rotates clockwise until being on the radius of C^* passing through the first unoccupied target, and finally moves radially to the target.
m_4	If $\lvert X\rvert = 3$ or ($\lvert X\rvert = 2$ and $\rho(R^+) = 2$), then all robots in X rotate clockwise; otherwise among r_1 and r_2 the farthest from L rotates clockwise.
m_5	The unique internal molecule radially moves along L until reaching a position consistent with a molecule forming the matter wrt $c(C^*)$.
m_6	The three internal molecules first rotate clockwise wrt to their center until the radii of C^* passing through their centers become orthogonal to the segments joining the two robots forming each molecule. Then, they radially move until reaching the right positioning in order to become part of the matter wrt $c(C^*)$.
m_7	If the matter is currently composed by at least three molecules, then let r_1, r_2 and r_3 be the first three robots met from p_1, p_2, and p_3, respectively, in the clockwise direction. If $\rho(R \setminus \{r1, r2, r3\}) = 3$ then r_1, r_2 and r_3 rotate along C^* until creating three molecules. If $\rho(R \setminus \{r1, r2, r3\}) \neq 3$, then the robot among r_1, r_2, and r_3 closest to the successive one in the clockwise direction rotates along C^* until forming a new molecule.
m_8	This task moves all the molecules formed by means of task T_7 to grow the matter. The molecule closest in the clockwise direction to the first available position of the last level of the matter not yet filled, moves there clockwise while possibly rotating wrt its center.

Table 3: Moves associated with tasks.

each molecule. Then, they radially move until reaching the right positioning in order to become part of the matter wrt $c(C^*)$.

Once T_5 or T_6 terminate, the matter is suitably initialized and the configuration admits a symmetricity of either 1 or 3, respectively.

Task T_7. This task forms new molecules on C^* (sub-problem Forming Molecules 2 (*FM2*)) that afterward are moved by T_8 so as to make the matter growing (see, eg Fig.s 4 and 5). If the matter is currently composed by at least three molecules, then let r_1, r_2 and r_3 be the first three robots met from p_1, p_2, and p_3, resp., in the clockwise direction. If the symmetricity of the configuration excluding r_1, r_2 and r_3 is 3, then r_1, r_2 and r_3 rotate along C^* until creating three molecules. If such a symmetricity is not 3 or the matter is currently composed by less than three molecules, then the robot on C^* closest to the successive one in the clockwise direction rotates along C^* until forming a new molecule.

Task T_8. This task moves all the molecules formed on C^* by means of tasks T_4 or T_7 to grow the matter (sub-problem Moving Molecules (*MM*)), see, eg Fig. 4. The molecule closest in the clockwise direction to the first available position of the last level of the matter not yet filled, moves there clockwise while possibly rotating wrt its center.

Task T_9. It refers to the requirement of letting molecules to detect the matter has been formed, hence no more movements are required (sub-problem Matter Done (*MD*)). Clearly, only *nil* movements are allowed and it is not possible to switch to any other task.

3.3 Running example

In this section, we show how robots correctly detect the task to perform. According to the definitions of P_i given in Eq. 1, in the `Compute` phase, each robot evaluates – wrt the perceived configuration – the preconditions starting from P_9 and proceeding in the reverse order until a true precondition is found. In case all preconditions $\texttt{pre}_9, \texttt{pre}_8, \dots, \texttt{pre}_2$ are evaluated false, then task T_1, whose precondition is simply `true`, is performed. It follows that the algorithm satisfies Prop_3.

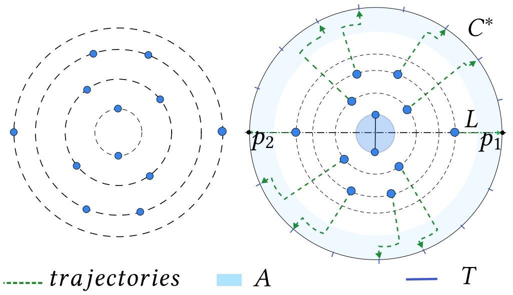

Figure 2: Configurations belonging to tasks T_1 (left) and T_3 (right). Relative distances are reduced for space constraints.

The initial configuration in Fig. 2.left is such that $\rho(R) = 2$ and belongs to T_1. In fact, since there are no molecules formed, `Mf`, `M2`, `M1`, `nM3`, `nM2`, `nM1`, and `cM` are false, that is the configuration is not in $T_9, \dots T_3$, respectively. Concerning `iM3″`, circle C_R contains only two robots, hence the predicate is false. Concerning `iM3′`, the distance from $c(R)$ detects 6 robots, hence the predicate is false too, that is the configuration is not in T_2 and then belongs to T_1. During T_1 the two most internal robots move toward each other according to m_1, hence the same considerations as above hold until their distance reduces to D and a molecule is formed, see Fig. 2.right.

The reached configuration in Fig. 2.right belongs to T_3. In fact, `Mf` is clearly false, that is the configuration is not in T_9. As `FarC` is false, that is `FarC` is false (see Table 1), then `M2`, `M1`, `nM3`, `nM2`, and `nM1` are false, that is the configuration in not in $T_8, \dots T_4$, respectively. Concerning `iM3″`, circle C_R contains only two robots, hence the predicate is false. Concerning `iM3′`, the distance from $c(R)$ detects 6 robots, hence the predicate is false too. Since $|Mol| = 1$ and not all the robots are on C^* then `cM` is true, that is the configuration belongs to T_3. In Fig. 2 also the trajectories traced by the robots are shown during T_3 and the above Boolean values hold until the

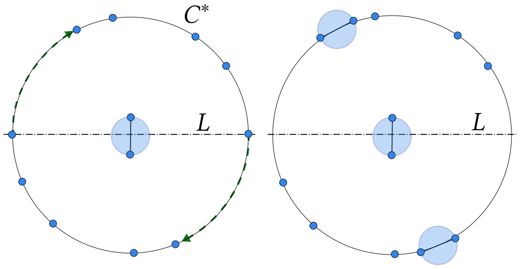

Figure 3: Configurations in tasks T_4 (left) and T_5 (right).

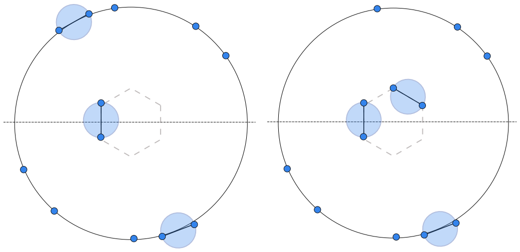

Figure 4: Two successive configurations belonging to task T_8.

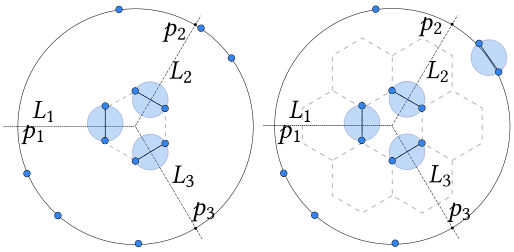

Figure 5: Configurations in tasks T_7 (left) and T_8 (right).

last robot reaches C^*. In particular, for iM3′ it is possible that at some point the distance from $c(R)$ identifies exactly three robots, however, in that case $\rho(R') < 3$.

The reached configuration in Fig. 3.left belongs to T_4. In fact, here $Mat = \emptyset$, hence Mf, M2, M1, are false and the configuration is not in T_9, T_8, nor T_7. As $|Mol| = 1$ then nM3 and nM2 are false, that is the configuration does not belong to T_6 nor T_5. Since $|Mat| = 0$, FarC is true, $|Mol'| = 1$ and there are no molecules on C^* then nM1 is true and the configuration is in T_4. The corresponding move m_4 makes robots on p_1 and p_2 rotate clockwise on C^* until forming two molecules. During the movements, it is possible that one molecule appears before the other but this does not affect the truth value of the above predicates.

The reached configuration in Fig. 3.right belongs to T_5. In fact, here $Mat = \emptyset$, hence Mf, M2, M1, are false and the configuration is not in T_9, T_8, nor T_7. As $|Mol| = 3$ then nM3 is false, that is the configuration does not belong to T_6. Since $|Mat| = 0$, FarC is true, $|Mol| = 3$ with 1 or 2 molecules on C^* and 1 internal, and the number of molecules on C^* is no more than $\rho(R)$ then nM2 is true and the configuration is in T_5. Here the internal molecule radially move along L reaching the side of an hexagon of side D centered in $c(C^*)$, hence making $|Mat| = 1$. During the movements the truth value of the above predicates is not affected. The reached configuration in Fig. 4 belongs to T_8. In fact, here Mf is false, that is the configuration is not in T_9. Since $|Mat| > 0$, FarC is true, $|Mol \setminus Mat| = 2$ and the matter does not admit a rotation, then M2 holds and the configuration is in T_8. Move m_8 involves the two external molecules, one by one, and leads them to be part of the matter. During the movements and after the first molecule arrives, $|Mol \setminus Mat| = 1$ and $\rho(R \setminus \{r_1, r_2, r_3\}) = 1$, hence the truth value of the above predicates is not affected. The reached configuration in Fig. 5 belongs to T_7. In fact, here Mf is false, that is the configuration is not in T_9. $|Mol| \setminus |Mat| = 0$, that is M2 is false the configuration is not in T_8. Since $|Mat| > 0$, $|Mol \setminus Mat| < 3$ and FarC is true then M1 holds and the configuration is in T_7. This holds also during the movement of the robots.

It is easy to see that the reached configuration belongs again to T_8 and by alternating with T_7 the final configuration in Fig. 1.left is achieved. According to precondition *MF*, the final configuration belongs to Task T_9, where only the *nil* movement is performed.

The reached configuration admits symmetricity 1 whereas the initial configuration of Fig. 2.left has symmetricity 2. According to Theorem 2.1, this is possible since $\rho(\mu) = 2$. Contrarily, within OBLOT, the disposal of the robots as specified by the defined matter could not be achieved. However, when each molecule in $\mathcal{M}$ is constituted by a single robot, then MOBLOT reduces to OBLOT.

4 CONCLUSION

In this paper, we have proposed MOBLOT, a new theoretical model in the context of swarm robotics that extends OBLOT. MOBLOT concerns two levels of computational entities: robots and molecules. Robots can be very weak entities like in the OBLOT model, although here they can be heterogeneous; molecules are more complex entities with an extent. Robots and molecules are guided by their respective distributed algorithms: the former is used to form molecules, the latter to assemble molecules to obtain some complex structure, the matter. Once the matter is formed, a third algorithm could even be used to rearrange (self-reconfigure) the molecules' positions to get a different shape for the matter.

We have proven that there is some necessary condition for forming the matter. This condition states that the symmetricity of the initial configuration of robots must divide either the symmetricity of some molecule, or the symmetricity of the matter to be formed. Interestingly, this implies that the matter could be formed even when the symmetricity of the input configuration is unrelated to that of the matter (in such cases, the molecules play a decisive role).

There are many directions for future research in the proposed model. The most obvious one is to investigate about a (complete) characterization of the solvability of the matter formation problem according to the assumed capabilities for robots and molecules. Others concern the formalization of possible self-reconfigurable matter problems, as well problems related to the matter movement.

ACKNOWLEDGMENTS

The work has been supported in part by the Italian National Group for Scientific Computation GNCS-INdAM.

REFERENCES

[1] Chrysovalandis Agathangelou, Chryssis Georgiou, and Marios Mavronicolas. 2013. A Distributed Algorithm for Gathering Many Fat Mobile Robots in the Plane. In *Proc. 32nd ACM Symp. on Principles of Distributed Computing (PODC)*.

[2] Hossein Ahmadzadeh, Ellips Masehian, and Masoud Asadpour. 2016. Modular Robotic Systems: Characteristics and Applications. *J. Intell. Robotic Syst.* 81, 3-4 (2016), 317–357. https://doi.org/10.1007/s10846-015-0237-8

[3] Michael Amir and Alfred M Bruckstein. 2019. Minimizing Travel in the Uniform Dispersal Problem for Robotic Sensors. In *Proc. 18th Int.'l Conf. on Autonomous Agents and MultiAgent Systems (AAMAS)*. International Foundation for Autonomous Agents and Multiagent Systems, 113–121.

[4] Sruti Gan Chaudhuri and Krishnendu Mukhopadhyaya. 2015. Leader election and gathering for asynchronous fat robots without common chirality. *J. Discrete Algorithms* 33 (2015), 171–192.

[5] Anders Lyhne Christensen. 2012. *Self-Reconfigurable Robots - An Introduction. Artif. Life* 18, 2 (2012), 237–240. https://doi.org/10.1162/artl_r_00061

[6] Serafino Cicerone, Alessia Di Fonso, Gabriele Di Stefano, and Alfredo Navarra. 2021. Arbitrary Pattern Formation on Infinite Regular Tessellation Graphs. In *Proc. 22nd Int.'l Conf. on Distributed Computing and Networking (ICDCN)*. ACM, New York, NY, USA, 56–65. https://doi.org/10.1145/3427796.3427833

[7] Serafino Cicerone, Gabriele Di Stefano, and Alfredo Navarra. 2016. Asynchronous Embedded Pattern Formation without Orientation. In *Proc. 30th Int.'l Symp. on Distributed Computing (DISC) (LNCS, Vol. 9888)*. Springer, 85–98.

[8] Serafino Cicerone, Gabriele Di Stefano, and Alfredo Navarra. 2018. Gathering of robots on meeting-points: feasibility and optimal resolution algorithms. *Distributed Computing* 31, 1 (2018), 1–50.

[9] Serafino Cicerone, Gabriele Di Stefano, and Alfredo Navarra. 2019. Asynchronous Arbitrary Pattern Formation: the effects of a rigorous approach. *Distributed Computing* 32, 2 (2019), 91–132.

[10] Serafino Cicerone, Gabriele Di Stefano, and Alfredo Navarra. 2019. Asynchronous Robots on Graphs: Gathering. In *Distributed Computing by Mobile Entities, Current Research in Moving and Computing*, Paola Flocchini, Giuseppe Prencipe, and Nicola Santoro (Eds.). LNCS, Vol. 11340. Springer, 184–217.

[11] Serafino Cicerone, Gabriele Di Stefano, and Alfredo Navarra. 2019. Embedded pattern formation by asynchronous robots without chirality. *Distributed Computing* 32, 4 (2019), 291–315.

[12] Serafino Cicerone, Gabriele Di Stefano, and Alfredo Navarra. 2021. Gathering robots in graphs: The central role of synchronicity. *Theor. Comput. Sci.* 849 (2021), 99–120. https://doi.org/10.1016/j.tcs.2020.10.011

[13] Mark Cieliebak, Paola Flocchini, Giuseppe Prencipe, and Nicola Santoro. 2012. Distributed Computing by Mobile Robots: Gathering. *SIAM J. on Computing* 41, 4 (2012), 829–879.

[14] Jurek Czyzowicz, Leszek Gasieniec, and Andrzej Pelc. 2009. Gathering few fat mobile robots in the plane. *Theor. Comput. Sci.* 410, 6-7 (2009), 481–499.

[15] Gianlorenzo D'Angelo, Mattia D'Emidio, Shantanu Das, Alfredo Navarra, and Giuseppe Prencipe. 2020. Asynchronous Silent Programmable Matter Achieves Leader Election and Compaction. *IEEE Access* 8 (2020), 207619–207634.

[16] Gianlorenzo D'Angelo, Mattia D'Emidio, Shantanu Das, Alfredo Navarra, and Giuseppe Prencipe. 2020. Leader Election and Compaction for Asynchronous Silent Programmable Matter. In *Proc. 19th Int.'l Conf. on Autonomous Agents and Multiagent Systems (AAMAS)*. International Foundation for Autonomous Agents and Multiagent Systems, 276–284.

[17] Joshua J. Daymude, Kristian Hinnenthal, Andréa W. Richa, and Christian Scheideler. 2019. Computing by Programmable Particles. In *Distributed Computing by Mobile Entities, Current Research in Moving and Computing*, Paola Flocchini, Giuseppe Prencipe, and Nicola Santoro (Eds.). Lecture Notes in Computer Science, Vol. 11340. Springer, 615–681. https://doi.org/10.1007/978-3-030-11072-7_22

[18] Zahra Derakhshandeh, Robert Gmyr, Thim Strothmann, Rida A. Bazzi, Andréa W. Richa, and Christian Scheideler. 2015. Leader Election and Shape Formation with Self-organizing Programmable Matter. In *DNA Computing and Molecular Programming - 21st International Conference, DNA 21, Boston and Cambridge, MA, USA, August 17-21, 2015. Proceedings (Lecture Notes in Computer Science, Vol. 9211)*, Andrew Phillips and Peng Yin (Eds.). Springer, 117–132. https://doi.org/10.1007/978-3-319-21999-8_8

[19] Ioan Dumitrache and Monica Dragoicea. 2008. Agent-based Theory Applied in Mobile Robotics. *IFAC Proceedings Volumes* 41, 2 (2008), 13719 – 13724. https://doi.org/10.3182/20080706-5-KR-1001.02323 17th IFAC World Congress.

[20] Paola Flocchini, Giuseppe Prencipe, and Nicola Santoro. 2019. Moving and Computing Models: Robots. In *Distributed Computing by Mobile Entities, Current Research in Moving and Computing*, Paola Flocchini, Giuseppe Prencipe, and Nicola Santoro (Eds.). Lecture Notes in Computer Science, Vol. 11340. Springer, 3–14. https://doi.org/10.1007/978-3-030-11072-7_1

[21] Paola Flocchini, Giuseppe Prencipe, Nicola Santoro, and Peter Widmayer. 2008. Arbitrary pattern formation by asynchronous, anonymous, oblivious robots. *Theor. Comput. Sci.* 407, 1-3 (2008), 412–447.

[22] Toshio Fukuda and Yoshio Kawauchi. 1990. Cellular robotic system (CEBOT) as one of the realization of self-organizing intelligent universal manipulator. In *Proc. of the 1990 IEEE Int.'l Conf. on Robotics and Automation, Cincinnati, Ohio, USA, May 13-18, 1990*. IEEE, 662–667. https://doi.org/10.1109/ROBOT.1990.126059

[23] Yonghwan Kim, Yoshiaki Katayama, and Koichi Wada. 2020. Pairbot: A Novel Model for Autonomous Mobile Robot Systems Consisting of Paired Robots. arXiv:2009.14426 [cs.DC]

[24] Zhiqiang Liu, Yukiko Yamauchi, Shuji Kijima, and Masafumi Yamashita. 2018. Team assembling problem for asynchronous heterogeneous mobile robots. *Theor. Comput. Sci.* 721 (2018), 27–41. https://doi.org/10.1016/j.tcs.2018.01.009

[25] Giuseppe Prencipe. 2019. Pattern Formation. In *Distributed Computing by Mobile Entities, Current Research in Moving and Computing*, Paola Flocchini, Giuseppe Prencipe, and Nicola Santoro (Eds.). Lecture Notes in Computer Science, Vol. 11340. Springer, 37–62. https://doi.org/10.1007/978-3-030-11072-7_3

[26] Michael Rubenstein, Christian Ahler, and Radhika Nagpal. 2012. Kilobot: A low cost scalable robot system for collective behaviors. In *IEEE Int.'l Conf. on Robotics and Automation (ICRA)*. IEEE, 3293–3298. https://doi.org/10.1109/ICRA.2012.6224638

[27] Erol Sahin. 2004. Swarm Robotics: From Sources of Inspiration to Domains of Application. In *Swarm Robotics, SAB 2004 Int.'l Workshop, Santa Monica, CA, USA, July 17, 2004, Revised Selected Papers (Lecture Notes in Computer Science, Vol. 3342)*, Erol Sahin and William M. Spears (Eds.). Springer, 10–20. https://doi.org/10.1007/978-3-540-30552-1_2

[28] Ichiro Suzuki and Masafumi Yamashita. 1999. Distributed Anonymous Mobile Robots: Formation of Geometric Patterns. *SIAM J. Comput.* 28, 4 (1999), 1347–1363.

[29] Thadeu Tucci, Benoît Piranda, and Julien Bourgeois. 2018. A Distributed Self-Assembly Planning Algorithm for Modular Robots. In *Proc. of the 17th Int.'l Conf. on Autonomous Agents and MultiAgent Systems (AAMAS)*, Elisabeth André, Sven Koenig, Mehdi Dastani, and Gita Sukthankar (Eds.). International Foundation for Autonomous Agents and Multiagent Systems Richland, SC, USA / ACM, 550–558.

[30] M. Yim, W. Shen, B. Salemi, D. Rus, M. Moll, H. Lipson, E. Klavins, and G. S. Chirikjian. 2007. Modular Self-Reconfigurable Robot Systems [Grand Challenges of Robotics]. *IEEE Robotics Automation Magazine* 14, 1 (2007), 43–52.