

## Gathering of Robots on Anonymous Grids without multiplicity detection

Gianlorenzo D'Angelo, Gabriele Di Stefano, Ralf Klasing, Alfredo Navarra

► **To cite this version:**

Gianlorenzo D'Angelo, Gabriele Di Stefano, Ralf Klasing, Alfredo Navarra. Gathering of Robots on Anonymous Grids without multiplicity detection. 19th International Colloquium on Structural Information and Communication Complexity (SIROCCO 2012), Jun 2012, Reykjavík, Iceland. Springer, 7355, pp.327-338, 2012, Lecture Notes in Computer Science. <10.1007/978-3-642-31104-8\_28>. <hal-00728988>

**HAL Id: hal-00728988**

**<https://hal.inria.fr/hal-00728988>**

Submitted on 7 Sep 2012

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Gathering of Robots on Anonymous Grids without Multiplicity Detection<sup>\*</sup>

Gianlorenzo D'Angelo<sup>1</sup>, Gabriele Di Stefano<sup>2</sup>, Ralf Klasing<sup>3</sup>, and  
Alfredo Navarra<sup>4</sup>

<sup>1</sup> MASCOTTE project CNRS-INRIA-UNS France. [gianlorenzo.d.angelo@inria.fr](mailto:gianlorenzo.d.angelo@inria.fr)

<sup>2</sup> Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica,  
Università degli Studi dell'Aquila, Italy. [gabriele.distefano@univaq.it](mailto:gabriele.distefano@univaq.it)

<sup>3</sup> CNRS / LaBRI / Université Bordeaux 1, France. [klasing@labri.fr](mailto:klasing@labri.fr)

<sup>4</sup> Dipartimento di Matematica e Informatica, Università degli Studi di Perugia, Italy.  
[alfredo.navarra@unipg.it](mailto:alfredo.navarra@unipg.it)

**Abstract.** The paper studies the gathering problem on grid networks. A team of robots placed at different nodes of a grid, have to meet at some node and remain there. Robots operate in Look-Compute-Move cycles; in one cycle, a robot perceives the current configuration in terms of occupied nodes (Look), decides whether to move towards one of its neighbors (Compute), and in the positive case makes the computed move instantaneously (Move). Cycles are performed asynchronously for each robot. The problem has been deeply studied for the case of ring networks. However, the known techniques used on rings cannot be directly extended to grids. Moreover, on rings, another assumption concerning the so-called *multiplicity detection* capability was required in order to accomplish the gathering task. That is, a robot is able to detect during its Look operation whether a node is empty, or occupied by one robot, or occupied by an undefined number of robots greater than one.

In this paper, we provide a full characterization about gatherable configurations for grids. In particular, we show that in this case, the multiplicity detection is not required. Very interestingly, sometimes the problem appears trivial, as it is for the case of grids with both odd sides, while sometimes the involved techniques require new insights with respect to the well-studied ring case. Moreover, our results reveal the importance of a structure like the grid that allows to overcome the multiplicity detection with respect to the ring case.

## 1 Introduction

In the field of robot based computing systems, one of the most popular problems is certainly the *gathering*. A pool of robots, initially situated at various locations, have to gather at the same place (not determined in advance) and remain there. Many variants of the problem have attracted the interest of numerous researchers (see e.g., [1, 2] and references therein). In this paper, we consider the case of

---

<sup>\*</sup> Research supported by the LaBRI under the “Project émergent” program.

anonymous grid networks where anonymous, asynchronous and oblivious robots can move according to the so-called Look-Compute-Move cycles [3]. In each cycle, a robot takes a snapshot of the current global configuration (Look), then, based on the perceived configuration, decides either to stay idle or to move to one of its adjacent nodes (Compute), and in the latter case it makes an instantaneous move to this neighbor (Move). Cycles are performed asynchronously for each robot. This means that the time between Look, Compute, and Move operations is finite but unbounded, and is decided by the adversary for each robot. Hence, robots may move based on significantly outdated perceptions. Robots are oblivious, i.e., they do not have any memory of past observations. Thus, the target node (which is either the current position of the robot or one of its neighbors) is decided by the robot during a Compute operation solely on the basis of the location of other robots perceived during the Look operation. Robots are anonymous and execute the same deterministic algorithm. They cannot leave any marks at visited nodes, nor send any messages to other robots. We remark that the Look operation provides the robots with the entire grid configuration concerning occupied nodes. That is, a robot perceives whether a node of the grid is occupied or not, but it cannot distinguish how many robots reside on an occupied node.

**Related Work and Our Results.** The problem of making mobile entities meet on graphs [3–6] or open spaces [1, 7, 8] has been extensively studied in the last decades. When only two robots are involved, the problem is usually referred to as the *rendezvous* [5, 9–11]. Under the Look-Compute-Move model, many problems have been addressed, like the *graph exploration* and the *perpetual graph exploration* [12–15], while the rendezvous problem has been proven to be unfeasible on rings [3].

Concerning the gathering under the Look-Compute-Move model, much work has been done in the last years for the ring topology. It has been proven that the gathering is unsolvable if the robots are not empowered by the so-called *multiplicity detection* capability [3], either in its *global/strong* or *local/weak* version. In the former type, a robot is able to perceive whether any node of the network is occupied by a single robot or more than one (i.e., a *multiplicity* occurs). In the latter type, a robot is able to perceive the multiplicity only if it is part of it.

Using the global multiplicity detection, different types of configurations have required different approaches. In particular, periodicity and symmetry arguments have been exploited. In a ring, a configuration is called *periodic* if it is invariable under non-trivial (i.e., non-complete) rotations. A configuration is called *symmetric* if the ring has a geometrical *axis of symmetry* that reflects single robots into single robots, multiplicities into multiplicities, and empty nodes into empty nodes. In [3], it is proven that, even with the global multiplicity detection, the gathering is unsolvable for two robots, for periodic configurations and for those symmetric configurations where the axis of symmetry passes through two edges. Then, several algorithms have been proposed for different kinds of initial configurations, in detail: for the case of odd number of robots and that of asymmetric configurations [3], for symmetric configurations with an even number of robots greater than 18 [16], and for 4 and 6 robots [17, 18].

Using the local multiplicity detection in a ring, in [19] it is shown that a configuration is gatherable if  $k < \lfloor \frac{n}{2} \rfloor$ , while in [20], the case where  $k$  is odd and smaller than  $n - 5$  is studied, where  $n$  and  $k$  are the number of nodes and robots, respectively. The remaining cases are still open.

In this paper, we fully characterize the gathering on grids. We show that the multiplicity detection capability is not needed. In particular, we show that even if the global multiplicity detection is assumed, a configuration is ungatherable only if it is periodic (i.e., the same view can be obtained by rotating the grid around its geometric center of an angle smaller than 360 degrees) on a grid with at least an even side, or it is symmetric with the axis of symmetry passing through edges. For all the other cases, we provide a gathering algorithm which does not require any multiplicity detection except for configurations on  $2 \times 2$  grids with three robots where the local multiplicity detection would be helpful.

To our knowledge, the grid topology is the least structured class of graphs that permits to avoid the multiplicity detection assumption. Moreover, it is worth mentioning that in our solutions many robots can move concurrently, instead of just one or two as it was for the ring case.

## 2 Definitions and Notation

We consider an anonymous and undirected grid of  $m \times n$  nodes, with  $m \geq n$ . Initially, each node is occupied by at most one robot. The total number of robots is denoted by  $k$ . During a Look operation, a robot perceives the relative locations on the grid of occupied nodes, regardless of the number of robots on a node.

The current configuration of the system can be described in terms of the view of a robot  $r$  which is performing the Look operation at the current moment. We denote a configuration seen by  $r$  as an  $m \times n$  matrix  $M$  on elements in the set  $\{0, 1\}$ . Value 0 represents an empty node, and 1 represents an occupied node. Note that, if one node is occupied by more than one robot, it is not perceived by the robots, even if they reside on such a node. Since the grid is anonymous and undirected, each robot can perceive the current configuration with respect to different rotations and reflections leading to any view of the grid satisfying the  $m \times n$  dimension. In particular, when  $m = n$  each of the 4 rotations and 4 reflections provides a feasible view.

**Definition 1.** *A configuration is periodic if it is invariant with respect to rotations of 90, 180, or 270 degrees, where the rotation point coincides with the geometric center of the grid.*

**Definition 2.** *A configuration is symmetric if it is invariant after a reflection with respect to a vertical, horizontal, or diagonal (in case of square grids) axis passing through the geometric center of the grid.*

## 3 Gathering algorithms

In this section, we distinguish among different cases concerning the grid structure. In particular, if the grid has both sides odd, the gathering is easily solvable.

If only one side is odd, there are some ungatherable cases. However, the impossibility results do not depend on the assumed multiplicity detection capability. If both sides are even, the gathering strategy relies on the multiplicity detection only if the input grid has size  $2 \times 2$  and there are three robots, otherwise there is no need of such a capability.

### 3.1 Odd $\times$ odd grids

This case is trivially solvable, in fact:

**Theorem 1.** *Configurations on odd  $\times$  odd grids are always gatherable.*

*Proof.* In odd  $\times$  odd grids, a robot can always detect, during its Look operation, the central node of the grid  $M[\lceil \frac{m}{2} \rceil, \lceil \frac{n}{2} \rceil]$ , regardless of its possible view. This means that all the robots can move toward the center, concurrently.  $\square$

### 3.2 Odd $\times$ even grids

In this case, the gathering is not always feasible. In fact, similarly to the ring case on periodic or symmetric configurations of type edge-edge [3], we can prove:

**Theorem 2.** *If a configuration  $C$  is periodic, or symmetric with respect to an axis passing through edges (i.e., dividing the grid into two halves from the even side), then  $C$  is ungatherable.*

In what follows, we assume that the starting configuration does not belong to the ungatherable configurations specified by the above theorem. Then, we provide an algorithm achieving the gathering without multiplicity detection in all the remaining cases. The idea is to distinguish among the two nodes that are the central nodes of the odd borders of the grid. If  $m$  ( $n$ , resp.) is odd, then the two mentioned nodes are given by positions  $M[1, \lceil \frac{m}{2} \rceil]$  and  $M[n, \lceil \frac{m}{2} \rceil]$  ( $M[\lceil \frac{n}{2} \rceil, 1]$  and  $M[\lceil \frac{n}{2} \rceil, m]$ , resp.). The line connecting those two nodes will be denoted as the NS line. One of the two extreme nodes on the NS line will be the place where the gathering is finalized, eventually. In order to select the gathering node, a robot considers the line passing through the central edges of the even sides of the grid (denoted as the EW line) dividing the grid into two halves. The idea is to distinguish a north and a south part among the two halves and the gathering node will be the one in the north half. The north is the half with more nodes occupied by robots, if any. If the number of occupied nodes in the two halves is the same, then some more computations are required (see next paragraph). In both cases, the robots move from the south to the north until all the robots will be in the north part. Note that, during such a stage, if multiplicities are created in the south, then the number of occupied nodes decreases with respect to the north part. If multiplicities are created in the north, it means that a robot has moved from the south to the north part, still preserving the required distinction.

In order to distinguish the north from the south in the case of configurations with the same number of robots among the two halves obtained by the EW line, a

robot associates to each configuration  $C$  a binary string as follows. Starting from each corner of the grid, and proceeding in the direction parallel to the NS line, a robot records the elements of  $M$  row by row, or column by column (according to the direction specified by the NS line). Once it has computed the four strings, it associates to  $C$  the lexicographically largest one. For instance, starting from corner  $M[1,1]$ , and assuming  $m$  odd, the corresponding binary string would be composed by the sequence  $M[1,1], M[2,1], \dots, M[n,1], M[1,2], \dots, M[n,2], M[1,m], \dots, M[n,m]$ .

**Lemma 1.** *Let  $C$  be a gatherable configuration, then, among the four possible strings coming from a robot view of the input grid, at most two strings can be the lexicographically largest ones. If there are two largest strings, then they represent the views of  $C$  starting from two symmetric corners with respect to the NS line.*

*Proof.* If the equal strings correspond to the view of  $C$  starting from two symmetric corners with respect to the EW line, then  $C$  would be symmetric with respect to the EW line. In fact, from Definition 2, this would correspond to a reflection of the grid with respect to the EW line. But, from Theorem 2, it would imply that  $C$  is ungatherable. If the equal strings correspond to the view of  $C$  starting from two corners residing on one of the two diagonals of the grid, then  $C$  would be periodic. In fact, from Definition 1, this would correspond to a rotation of the grid of 180 degrees, again despite  $C$  being gatherable. Then, no more than two strings can be equal as otherwise one of the above situations would occur. It follows that either the four strings are all different among themselves, or there are two pairs of equal strings, one of which corresponds to the lexicographically largest ones. Moreover, both correspond to the view of  $C$  starting from two symmetric corners with respect to the NS line.  $\square$

From the above lemma, we define the *gathering node* as the one residing on the same odd side where the corner(s) providing the lexicographically largest string resides. Moreover, the gathering node will determine also the directions along the NS line: We say that the gathering node resides on the north pole.

**Theorem 3.** *Configurations on odd $\times$ even grids that are aperiodic and do not admit an axis of symmetry passing through edges are always gatherable.*

*Proof.* Once the gathering node has been unambiguously identified by a robot during its Compute operation, if the robot resides on the half grid where the south pole is, with respect to the EW line, then it moves towards the north. Note that, each time a robot in the southern half of the grid performs such a movement, the gathering node cannot change. In fact, two cases can occur: 1) the number of occupied nodes decreases in the southern part of the grid, either because a robot moves to the northern part or because a multiplicity is created; 2) the string associated to the corners in the south are decreasing due to the robots' movements, and hence the corresponding strings defining the current configuration starting from the northern corners are increasing. This clearly leaves unchanged the direction on the NS line. Note that the corner to which the

lexicographically largest string was associated might change during the described process but the only option is the other corner on the same odd side of the original one, hence preserving the direction on the NS line. By keeping on moving in the described way, all the robots will reach the northern part, eventually. The case in which a subset of robots from a multiplicity move, increasing the number of occupied nodes, does not require any special treatment. In fact, since the initial configuration does not contain multiplicities, either the minimality of the number of robots in one half of the grid is preserved, or case 2) still ensures that the lexicographically largest string is associated to a corner in the north.

Once all the robots belong to one half of the grid, then they are allowed to move, during their Move operation, towards the gathering node. In fact, such a node is now well-defined and cannot change as the robots are not allowed to move to the other half of the grid.  $\square$

### 3.3 Even $\times$ even grids

In this section, we study the case of grids whose sides are both even. Also in this case, by Theorem 2, there are some configurations which are ungatherable, namely the periodic configurations and those configurations having a vertical or a horizontal axis of symmetry. We show that all the other cases are gatherable without any multiplicity detection, but for the case of  $2 \times 2$  grids.

**Theorem 4.** *Let us consider a  $2 \times 2$  grid with more than one node occupied. If the multiplicity detection is not allowed, then any configuration is ungatherable. If the local multiplicity detection is allowed, a configuration is gatherable if and only if it has three nodes occupied.*

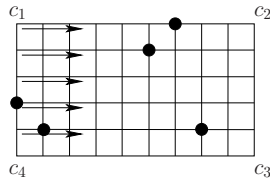
*Proof.* Clearly, a  $2 \times 2$  grid equals a ring of dimension four. Hence, any ungatherable configuration on a ring of four nodes is ungatherable on a  $2 \times 2$  grid. In particular, configurations with two or four nodes occupied are ungatherable even with the global multiplicity detection and configurations with three nodes occupied are ungatherable if the multiplicity detection is not allowed [3].

Finally, we show that a configuration is gatherable with the local multiplicity detection if three nodes are occupied. Consider the following algorithm:

1. move the robot in between the other two occupied nodes arbitrarily;
2. move the robot not in the multiplicity towards the other occupied node.  $\square$

Hence, the remaining gatherable configurations are the aperiodic, asymmetric, and those with only one axis of symmetry passing through the diagonal of a square grid of dimensions larger than  $2 \times 2$ . We refer to all such configurations as the set EG (Even-Gatherable). In Theorem 5, we will show that all the configurations in EG are indeed gatherable without any multiplicity detection.

In the following, we first assume that at least one node on the border of the grid is occupied. Then, in the proof of Theorem 5, we will show how to extend the given strategies to the general case. First, we give some definitions about the “reading” of grid configurations needed for the subsequent proofs.



**Fig. 1.** Case of a  $10 \times 6$  grid. The arrows indicate the horizontal direction of the reading from corner  $c_1$ , it gives  $(6, 8, 14, 10, 5, 12)$ . The other seven sequences read by the robots are:  $(3, 6, 20, 4, 9, 13)$  from  $c_1$  vertically,  $(3, 10, 24, 2, 5, 11)$  and  $(16, 1, 6, 26, 4, 2)$  from  $c_2$  horizontally and vertically, resp.,  $(12, 5, 10, 14, 8, 6)$  and  $(13, 9, 4, 20, 6, 3)$  from  $c_3$ ,  $(11, 5, 2, 24, 10, 3)$  and  $(2, 4, 26, 6, 1, 16)$  from  $c_4$ . The *minimal* sequence is  $(2, 4, 26, 6, 1, 16)$  and  $c = c_4$ .

Let us consider the eight sequences of distances (number of empty nodes) between occupied nodes obtained by traversing the grid starting from the four corners and proceeding towards the two possible directions (see, e.g. Fig. 1). Note that the two sequences associated to a corner occupied by some robot starts with 0. We associate for each corner the lexicographically smallest sequence between the two readings from such corner. Note that, in square grids such two sequences are always different, but for the two corners through which passes the possible axis of symmetry. In rectangular grids, these two sequences can be equal but we can distinguish one of them by assuming that if two sequences are equal, the one read in the direction of the smallest side is smaller than the other.

We define the *minimal* sequence as follows. If the configuration is symmetric, it is the smallest sequence between the two sequences associated to the two corners through which passes the axis of symmetry, otherwise it is the smallest among the four sequences associated to the four corners. Note that, under the assumption that the configuration does not fall into the hypothesis of Theorem 2, in any case there exists a minimal sequence which identifies a single corner, unambiguously. We denote the minimal sequence as  $C = (q_0, q_1, \dots, q_j)$  and by  $c$  the corner which it is associated to.

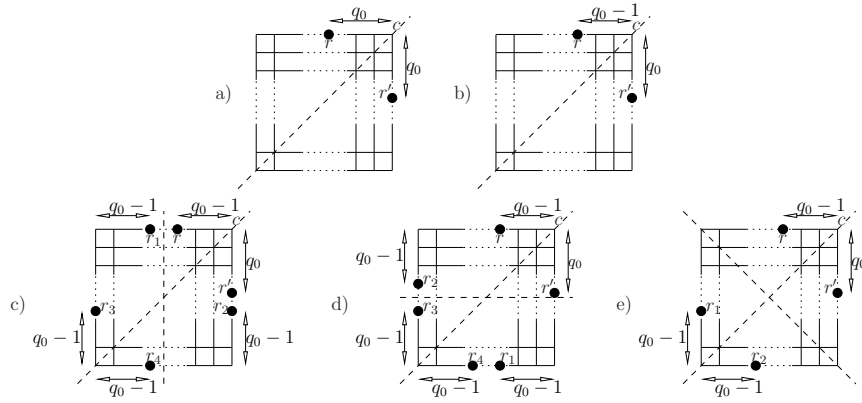
An important property of our gathering strategy that we are going to present is: *In all the movements used in the following results we do not allow a robot to move into a corner different from  $c$ .*

**Lemma 2.** *For any EG configuration with no corners occupied and at least one robot on the border there exists a strategy that leads to a configuration with exactly one corner occupied.*

*Proof.* If there are no corners occupied, the idea is to reduce  $q_0$  by moving towards  $c$  the robot (or the two robots, when the configuration is symmetric) on the border which is (are) closest to  $c$ . Note that, as we are assuming that there is at least a node on the border and that  $C$  is the minimal sequence, the robot (robots) involved in the movement towards  $c$  is (are) on the border.

In the case of symmetric configurations, we aim to move towards  $c$  the two symmetric robots on the border which are the closest ones to  $c$ . Let us denote





**Fig. 2.** Symmetric square grid with no corners occupied. Dashed lines represent axes of symmetry. a) original configuration; b) configuration once only robot  $r$  has moved; c) configuration with a possible vertical axis; d) configuration with a possible horizontal axis; e) configuration with a possible diagonal axis different from the original one.

these two robots by  $r$  and  $r'$ . First of all, we prove that if only one robot moves (let us assume  $r$ ), no symmetric configuration can be created and, moreover, there exists only one robot ( $r'$ ) at one allowed move from a symmetric configuration. We prove this by showing that contradicting such statement would imply that, in the initial configuration, the sequence associated to the corner on the axis different from  $c$  starts with  $q_0 - 1$  (see Fig. 2), which is a contradiction with respect to the minimality of  $C$ . Moving robot  $r$  may create three potential axes of symmetry which are, with respect to the drawing of Fig. 2: a vertical, horizontal, or diagonal axis different from the original one. Let us assume that there exists a vertical axis of symmetry (see Fig. 2c). In this case the presence of the vertical axis of symmetry and of robot  $r$  at distance  $q_0 - 1$  from  $c$ , implies the presence of robot  $r_1$  of Fig. 2c which is specular to  $r$  with respect to the vertical axis of symmetry. Note that, in this case,  $q_0 - 1 < \frac{n}{2}$  as otherwise  $r_1$  would be closer to  $c$  than  $r$ , a contradiction. Therefore,  $r_1$  is at distance  $q_0 - 1$  from the upper left corner of Fig. 2c. Now, since this robot did not move, the original diagonal axis of symmetry implies the presence of a robot  $r_2$  at distance  $q_0 - 1$  from the lower right corner of Fig. 2c. Again, the two axes imply the presence of robots  $r_3$  and  $r_4$ , both at distance  $q_0 - 1$  from the corner opposite to  $c$ , a contradiction to the minimality of  $C$ . The case of horizontal axis of symmetry is similar and it is shown in Fig. 2d. Let us now assume that there exists a diagonal axis of symmetry different from the original one (see Fig. 2e). Such axis implies the presence of robot  $r_1$  at distance  $q_0 - 1$  from the corner opposite to  $c$ . Moreover, the original axis implies the presence of robot  $r_2$  at distance  $q_0 - 1$  from the corner opposite to  $c$ . Again, this is a contradiction to the minimality of  $C$ . Similar arguments can be used to show that there cannot exist other robots besides  $r'$  at one allowed move from a symmetric configuration. Hence, in the

case of symmetric configurations, or asymmetric configurations at one allowed move from symmetry, the strategy leads to the occupation of  $c$ , possibly with a pending move towards  $c$ .

In the case of asymmetric configurations, first of all the robots check whether the configuration is at one step from a symmetric configuration belonging to  $\mathbf{EG}$  that can be obtained by the move performed by a robot (potentially corresponding to  $r'$ ) on the border towards the corner  $c$  that would be obtained when the axis of symmetry occurs. We recall that, in the case of symmetric configurations,  $c$  is defined as the corner whose associated sequence is the smallest one among the two corners lying on the axis of symmetry and not among all the corners. If such a symmetry cannot be established, then  $c$  corresponds to the minimal sequence and the algorithm proceeds by reducing  $q_0$  until it becomes 0, that is,  $c$  is occupied. In fact, if the initial configuration is asymmetric and  $q_0 > 1$ , then after reducing  $q_0$ , the obtained configuration is again asymmetric as the minimal sequence of the new configuration starts with  $q_0 - 1$  while any other sequence starts with at least  $q_0$ . When the initial configuration is asymmetric and  $q_0 = 1$ , after the move, the configuration might become symmetric but with one corner occupied. In conclusion, in any case the obtained configuration has the corner  $c$  occupied, and possibly one pending move towards  $c$ .  $\square$

Before showing the case of configurations in  $\mathbf{EG}$  with two corners occupied, we need to exclude all the configurations with exactly three occupied nodes, two of which are two corners that share a coordinate, and the other one is at one node apart from another corner. See the configuration in the middle of Fig. 3. We denote such configurations as the set  $3\mathbf{EG}2$ .

**Lemma 3.** *For any configuration in  $\mathbf{EG} \setminus 3\mathbf{EG}2$  with two corners occupied there exists a strategy that leads to a configuration with either exactly one corner occupied or exactly three corners occupied.*

*Proof.* If two corners are occupied, and the configuration is symmetric with the axis passing through the occupied corners, then one of them corresponds to  $c$ . We move the robot in the occupied corner which does not corresponds to  $c$  towards the other one, and we end up with the case of only one corner occupied.

The case where two corners are occupied, and the configuration is asymmetric or symmetric with the axis not passing through the occupied corners requires some more effort. In this case, it is risky to move the robots from the occupied corners, since if the adversary forces to move only one of them, we could not be able to recognize the possible move of the other robot which is still pending. Therefore, let  $d$  be the corner not occupied by a robot from which we read the minimal sequence  $D$ . We move towards  $d$  the first robot not in a corner that reduces  $D$ , and possibly, the symmetric one.<sup>5</sup> By repeating this strategy,  $d$  will be occupied by at least one robot, eventually. Arguments similar to those used in the proof of Lemma 2 can be used to show that no symmetries different from the original one can be created. It follows that for each step in which this strategy is

<sup>5</sup> Here we do not need to preserve the symmetry as it was necessary in Lemma 2.

applied,  $d$  remains the same. The final configuration has three corners occupied and possibly one pending move towards  $d$  of a robot not in a corner.  $\square$

Note that the configuration obtained after the strategy given in Lemma 3 is always in EG with possibly one pending move.

**Theorem 5.** *Aperiodic configurations on even $\times$ even grids larger than  $2\times 2$ , that do not admit an axis of symmetry passing through edges, are gatherable.*

*Proof.* First, we can restrict the set of possible grids as follows. Let us consider the minimal even $\times$ even sub-grid which is centered in the geometrical middle of the original grid and includes all the occupied nodes of it. Such minimal *wrapping* grid is still of type even $\times$ even and preserves the possible symmetry of the original one. Moreover, it always has at least an occupied node on the border. Our algorithm only uses such sub-grids and it never changes the size of it, i.e. it neither enlarges it by moving robots outside of it, nor it reduces it by moving the robots on the border inside. Therefore, in what follows, we can assume without loss of generality that a grid always has at least an occupied node in the border. However, the case in which the resulting wrapping grid is a  $2\times 2$  grid will be considered separately.

If no corners are occupied, we can apply Lemma 2 that leads to a configuration with one corner occupied with possibly one pending move towards  $c$ . However, such a move would have been performed also in the strategy used for the case of one corner occupied which is given in the following.

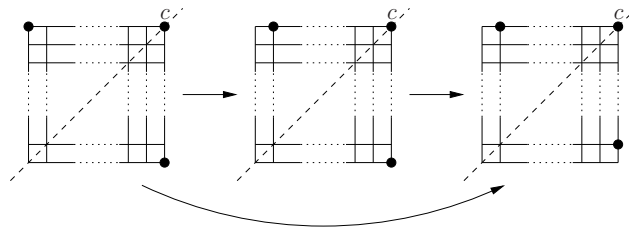
If only one corner is occupied, then it corresponds to  $c$ . In this case, all the robots move towards  $c$  by reducing the Manhattan distance to  $c$  and then achieving the gathering. We remind that the allowed movements are performed always without occupying any other corner than  $c$ .

In case two corners are occupied but the configuration is not in 3EG2, we apply the strategy of Lemma 3 and reach a configuration with one or three corners occupied. Also in this case there could be a pending move and, again, such a move would have been performed also in the strategy for one corner or three corners occupied.

If three corners are occupied, we move all the robots, but those in the corners, towards the corner that does not share any coordinate with the empty corner. This process finishes with a symmetric configuration with exactly three corners occupied. In this configuration,  $c$  is the corner on the axis of symmetry, and the other two robots move one step towards  $c$  either concurrently or alternately, until creating a configuration with only one corner occupied as shown in Fig. 3. Note that these final steps also solve the gathering for the configurations in 3EG2.

If four corners are occupied, we move the robot which occupies the corner farthest from  $c$  in an arbitrary direction, generating a configuration where only three corners are occupied.

It remains the case where the minimal wrapping even $\times$ even sub-grid which includes all the occupied nodes of the original grid has dimension  $2\times 2$ . As shown in Theorem 4, the configuration is not gatherable on this sub-grid without multiplicity detection. However, in the case of exactly three nodes occupied, we can



**Fig. 3.** Strategy from a symmetric square grid where exactly three corners are occupied and all the other nodes are empty to a configuration with only one occupied corner, possibly passing through configurations in 3EG2.

exploit the larger dimensions of the original grid in order to avoid the multiplicity detection. The cases of two or four nodes occupied clearly remain not gatherable. The strategy is then to move the robot on the corner of the  $2 \times 2$  grid which is in between the other two occupied corners towards the external row or column, arbitrarily. In doing so, we obtain the case where the minimal wrapping grid has dimension  $4 \times 4$  and no corners are occupied.  $\square$

## 4 Conclusion

We fully characterized the gathering in the Look-Compute-Move model on grids. We have shown that a configuration is ungatherable if and only if it is periodic on a grid with at least an even side, or it is symmetric with the axis passing through edges, or it is a  $2 \times 2$  grid. For all the other cases we provided a gathering algorithm which does not require any multiplicity detection. It would be of interest to investigate whether the grid topology is the least structured class of graphs that permits to avoid the multiplicity detection assumption.

## References

1. Degener, B., Kempkes, B., Langner, T., Meyer auf der Heide, F., Pietrzyk, P., Wattenhofer, R.: A tight runtime bound for synchronous gathering of autonomous robots with limited visibility. In: Proc. of the 23rd ACM Symp. on Parallelism in algorithms and architectures (SPAA). (2011) 139–148
2. Flocchini, P., Prencipe, G., Santoro, N., Widmayer, P.: Gathering of asynchronous robots with limited visibility. *Theor. Comput. Sci.* **337** (2005) 147–168
3. Klasing, R., Markou, E., Pelc, A.: Gathering asynchronous oblivious mobile robots in a ring. *Theor. Comput. Sci.* **390** (2008) 27–39
4. Bampas, E., Czyzowicz, J., Gasieniec, L., Ilcinkas, D., Labourel, A.: Almost optimal asynchronous rendezvous in infinite multidimensional grids. In: Proc. of the 24th Int. Symp. on Distributed Computing (DISC). Volume 6343 of Lecture Notes in Computer Science. (2010) 297–311
5. Dessmark, A., Fraigniaud, P., Kowalski, D., Pelc, A.: Deterministic rendezvous in graphs. *Algorithmica* **46** (2006) 69–96

6. Izumi, T., Izumi, T., Kamei, S., Ooshita, F.: Randomized gathering of mobile robots with local-multiplicity detection. In: Proc. of the 11th Int. Symp. on Stabilization, Safety, and Security of Distributed Systems (SSS). Volume 5873 of Lecture Notes in Computer Science. (2009) 384–398
7. Cord-Landwehr, A., Degener, B., Fischer, M., Hüllmann, M., Kempkes, B., Klaas, A., Kling, P., Kurras, S., Märten, M., Der Heide, F.M.A., Raupach, C., Swierkot, K., Warner, D., Weddemann, C., Wonisch, D.: A new approach for analyzing convergence algorithms for mobile robots. In: Proc. of the 38th Int. Colloquium on Automata, languages and programming (ICALP). Volume 6756 of Lecture Notes in Computer Science. (2011) 650–661
8. Prencipe, G.: Impossibility of gathering by a set of autonomous mobile robots. *Theor. Comput. Sci.* **384** (2007) 222–231
9. Alpern, S.: The rendezvous search problem. *SIAM J. Control Optim.* **33** (1995) 673–683
10. Chalopin, J., Das, S.: Rendezvous of mobile agents without agreement on local orientation. In: Proc. of the 37th Int. Colloquium on Automata, languages and programming (ICALP). Volume 6199. (2010) 515–526
11. Czyzowicz, J., Labourel, A., Pelc, A.: How to meet asynchronously (almost) everywhere. In: Proc. of the 21st Annual ACM-SIAM Symp. on Discrete Algorithms (SODA). (2010) 22–30
12. Blin, L., Milani, A., Potop-Butucaru, M., Tixeuil, S.: Exclusive perpetual ring exploration without chirality. In: Proc. of the 24th Int. Symp. on Distributed Computing (DISC). Volume 6343 of Lecture Notes in Computer Science., Springer (2010) 312–327
13. Devismes, S., Petit, F., Tixeuil, S.: Optimal probabilistic ring exploration by semi-synchronous oblivious robots. In: Proc. of the 16th Int. Colloquium on Structural Information and Communication Complexity (SIROCCO). Volume 5869 of Lecture Notes in Computer Science. (2009) 195–208
14. Flocchini, P., Ilcinkas, D., Pelc, A., Santoro, N.: Computing without communicating: Ring exploration by asynchronous oblivious robots. *Algorithmica* To appear.
15. Flocchini, P., Ilcinkas, D., Pelc, A., Santoro, N.: Remembering without memory: Tree exploration by asynchronous oblivious robots. *Theor. Comput. Sci.* **411**(14–15) (2010) 1583 – 1598
16. Klasing, R., Kosowski, A., Navarra, A.: Taking advantage of symmetries: Gathering of many asynchronous oblivious robots on a ring. *Theor. Comput. Sci.* **411** (2010) 3235–3246
17. D’Angelo, G., Di Stefano, G., Navarra, A.: Gathering of six robots on anonymous symmetric rings. In: Proc. of the 18th Int. Colloquium on Structural Information and Communication Complexity (SIROCCO). Volume 6796 of Lecture Notes in Computer Science. (2011) 174–185
18. Koren, M.: Gathering small number of mobile asynchronous robots on ring. *Zeszyty Naukowe Wydziału ETI Politechniki Gdańskiej. Technologie Informacyjne* **18** (2010) 325–331
19. Izumi, T., Izumi, T., Kamei, S., Ooshita, F.: Mobile robots gathering algorithm with local weak multiplicity in rings. In: Proc. of the 17th Int. Colloquium on Structural Information and Communication Complexity (SIROCCO). Volume 6058 of Lecture Notes in Computer Science. (2010) 101–113
20. Kamei, S., Lamani, A., Ooshita, F., Tixeuil, S.: Asynchronous mobile robot gathering from symmetric configurations. In: Proc. of the 18th Int. Colloquium on Structural Information and Communication Complexity (SIROCCO). Volume 6796 of Lecture Notes in Computer Science. (2011) 150–161