Anonymous Graph Exploration without Collision by Mobile Robots

Roberto Baldoni, François Bonnet, Alessia Milani, Michel Raynal

ANONYMOUS GRAPH EXPLORATION WITHOUT COLLISION BY MOBILE ROBOTS

R. BALDONI   F. BONNET   A. MILANI   M. RAYNAL

Anonymous Graph Exploration without Collision by Mobile Robots

R. Baldoni[*]   F. Bonnet[**]   A. Milani[***]   M. Raynal[****]

Systèmes communicants
Projet ASAP

Abstract: Considering autonomous mobile robots moving on a finite anonymous graph, this paper focuses on the *Constrained Perpetual Graph Exploration* problem ($CPGE$). That problem requires each robot to perpetually visit all the vertices of the graph, in such a way that no vertex hosts more than one robot at a time, and each edge is traversed by at most one robot at a time. The paper states an upper bound k on the number of robots that can be placed in the graph while keeping $CPGE$ solvability. To make the impossibility result as strong as possible (no more than k robots can be initially placed in the graph), this upper bound is established under a strong assumption, namely, there is an omniscient daemon that is able to coordinate the robots movements at each round of the synchronous system. Interestingly, this upper bound is related to the topology of the graph. More precisely, the paper associates with each graph a labeled tree that captures the paths that have to be traversed by a single robot at a time (as if they were a simple edge). The length of the longest of these labeled paths reveals to be the key parameter to determine the upper bound k on the number of robots.

Key-words: Distributed algorithm, Colllision, Mutual exclusion constraint, Robot, Synchronous message-passing system.

(Résumé : tsvp)

[*] Università di Roma "La Sapienza", Italy, baldoni@dis.uniroma1.it
[**] Campus de Beaulieu, 35042 Rennes Cedex, France Francois.Bonnet@irisa.fr
[***] Università di Roma "La Sapienza", Italy, Alessia.Milani@dis.uniroma1.it
[****] IRISA, Université de Rennes 1, Campus de Beaulieu, 35042 Rennes Cedex, France raynal@irisa.fr

Exploration avec contraintes de graphes par des robots

Résumé : Ce rapport présente un borne sur le nombre maximal de robots lorsque ceux-ci doivent visiter infiniment souvent chasque sommet d'un graphe connexe sous les contraintes d'exclusion mutuelle sur les arcs et les sommets du graphe.

Mots clés : Algorithme distribué, collision, contrainte d'exclusion mutuelle, robot, système synchrone.

1 Introduction

Graph exploration by robots   The graph exploration problem consists in making one or several mobile entities visit each vertex of a connected graph. The mobile entities are sometimes called agents or robots. In the following we use the word "robot". The exploration is perpetual if the robots have to revisit forever each vertex of the graph. Perpetual exploration is required when robots have to move to gather continuously evolving information or to look for dynamic resources (resources whose location changes with time). If nodes and edges have unique labels, the exploration is relatively easy to achieve.

The graph exploration problem becomes more challenging when the graph is anonymous (i.e., the vertices, the edges, or both have no label). In such a context, several bounds have been stated. They concern the period to complete a visit of the nodes (e.g. [5, 9, 11]), or the size of the robot memory necessary to explore a graph (e.g., it is proved in [7] that a robot needs $O(D \log d)$ bits of local memory in order to explore any graph of diameter D and maximum degree d). Impossibility results for one or more robots with bounded memory (computationally speaking, a robot is then a finite state automaton) to explore all graphs have been stated in [4, 13]. The major part of the results on graph exploration consider that the exploration is made by a single robot. Only very recently, the exploration of a graph by several robots has received attention. This is motivated by research for more efficient graph explorations, fault-tolerance, or the need to overcome impossibilities due to the limited capabilities of a single robot.

The constrained exploration problem   This paper considers an extension of the Perpetual Exploration problem, namely, the Constrained Perpetual Exploration problem ($CPGE$), and investigates the maximum number of robots that keep its solvability. In $CPGE$, each robot has to visit the complete graph infinitely many times, while avoiding several robots to collide in a same vertex or to concurrently traverse a same edge. These additional constraints are intended to abstract the problem of collision that robots may incur when moving within a short distance from each other or the necessity for the robots to access resources in mutual exclusion. (This mutual exclusion constraint has been considered in [10] in a robot movement problem in a grid.)

The result exposed in the paper concerns the upper bound k on the number of robots that keep the solvability of $CPGE$ problem in an anonymous connected graph. To obtain an impossibility result as strong as possible, the considered computation model is pretty strong, namely, the system is synchronous and the movement of the robots can be dictated by an omniscient entity. Not surprisingly, the upper bound k is related to the structure of the graph. (It is easy to see that if the graph is a chain, it is not possible for more than one robot to visit all the vertices while satisfying the mutual exclusion constraints on each vertex and each edge.)

Roadmap   The paper is made up of 6 sections. Section 2 presents the computation model, while Section 3 introduces the $CPGE$ problem. Section 4 presents related works. Then, Section 5 states and proves a bound on the maximal number of robots that can simultaneously solve the $CPGE$ problem. To that end, a reduction procedure from a graph to a labeled tree is introduced. This tree captures in a simple way the paths of the graph that have to be traversed by a single robot at a time (as if each of these paths was a simple edge). The length of the longest of these labeled paths then provides us with the upper bound on the number of robots. Finally, Section 6 concludes the paper.

2 Model

The system is made up of a finite connected undirected graph $G = (S, E)$ and a finite set R of robots such that $|R| \leq |S|$. The graph G is anonymous in the sense that the vertices have no identity. The robots have a common clock, and the time is divided into synchronous rounds [14]. At each round a robot can move from the vertex where it currently stays to a neighbor vertex, or stays at the same vertex. A move of a robot from a vertex to a neighbor vertex is done in one round. A configuration is an assignment of the robots to vertices such that there is at most one robot per vertex.

The moves of the robots are defined by an omniscient daemon whose computational power is a Turing machine. This omniscient daemon abstracts the capacity of the robots to globally synchronize their movements. Providing each robot with (1) a map of the graph, (2) the initial position of each robot, (3) the program executed by the daemon, allows each robot to simulate that omniscient daemon, and behaves accordingly.

Definition 1 *Given a robot a and a round r, $V(a, r)$ denotes the (single) vertex where the robot a is located at the beginning of round r.*

3   The Constrained Perpetual Exploration Graph Problem

As already indicated, the *Constrained Perpetual Exploration Problem* ($CPGE$) consists in designing an algorithm that makes each robot to visit infinitely often each vertex of the graph, in such a way that, at any round, no two robots stay on the same vertex, and during each round no two robots move on the same edge. That can be formally defined by the following three properties.

- Perpetual Exploration.
$$\forall v \in S : \ \forall a \in R : \ \ \{r \mid V(a,r) = v\} \text{ is not finite.}$$

  For any vertex v and any robot a, there are infinitely many rounds where a visits v.

- Vertex Mutual Exclusion.
$$\forall r \geq 0 : \ \forall (a,b) \in R \times R : \ \ (a \neq b) \quad \Rightarrow \quad V(a,r) \neq V(b,r).$$

  At the beginning of any round, no two robots are at the same vertex.

- Edge Mutual Exclusion.
$$\forall r \geq 0 : \ \forall (a,b) \in R \times R : \ \ (V(a, r+1) = V(b,r)) \Rightarrow (V(b, r+1) \neq V(a,r)).$$

  During a round, no two robots move on the same edge (i.e., they cannot exchange their positions).

4   Related work

On the initial assumptions   As already indicated, graph exploration is the process by which each vertex of a graph is visited by some entity. A great research effort on graph exploration by robots has been on the minimal assumptions (in terms of robots/network requirements) required to explore a graph (e.g., [3, 8]). Some works focus on robots endowed with a finite persistent storage and direct communication with other robots (e.g., [2]). Some works assume that each robot has the capability to see where other robots are currently placed (e.g., [6]). Some other works study how the knowledge of the map (graph) by the robots reduces the complexity of the exploration (e.g., [12]).

In our work, all these assumptions (on robots capabilities or initial knowledge) are captured by an omniscient daemon that govern the robot movements in the graph. This is motivated by the fact we want to state an impossibility result as strong as possible, i.e., with a particularly strong daemon.

On the type of graph exploration   Graph exploration is mainly divided into *perpetual* graph exploration (e.g., [5]) where the robots have to travel the graph infinitely often, and graph exploration *with stop* (e.g., [8]) where each robot has to eventually stop after having explored the graph. Some papers focus on the exploration of the graph by a single robot (e.g., [1, 2, 7]). Cooperative graph exploration by a team of mobile robots has also received attention (e.g., [3, 6, 10]). As an example of multi-robot exploration, it is shown in [6] that the minimum number of robots required to solve the exploration with stop of a ring of size n is $O(\log n)$ when the exploration is done by oblivious anonymous robots that move asynchronously.

Lower bounds have been established for the perpetual graph exploration problem (e.g., [5, 9]). These bounds concern the period necessary for a robot to complete the visit of the graph, assuming constraints either on the robots, or on the graph. Differently, the upper bound introduced in this paper concerns the maximum number of robots that can visit the graph infinitely often without ever colliding.

Constrained graph exploration As already noticed, The $CPGE$ problem defined in Section 3 is the perpetual graph exploration problem augmented with the mutual exclusion property on the vertices and the edges of the graph. These mutual exclusion constraints have been already stated and used [10] where the graphs considered are grids. The problem addressed in that paper is different from $CPGE$. More precisely, in [10], each robot has to visit some target vertices of the grid, and any two distinct robots have different targets. That paper establishes a lower bound on the time (number of rounds in a synchronous system) necessary to solve that problem and presents an optimal algorithm.

The problem of robots collision in also addressed in [15], where it is proposed a collision prevention algorithm for robots moving on the plane.

5 An Upper Bound on the Number of Robots

Given a graph G, let k be the maximal number of robots for which the $CPGE$ problem can be solved. The notion of problem solvability used here is (as already indicated) the one of an omniscient daemon that knows the graph and the current position of the robots, and defines for each round the behavior of each robot. As noticed in the Introduction, this very strong notion of problem solvability is used in order to have an impossibility result as strong as possible, namely, even with such a daemon it is not possible to solve $CPGE$ with more than k robots.

It is easy to see that the topology of a graph imposes naturally a bound on the number of robots that keep the solvability of $CPGE$. This is the case for the (previously cited) simple example of any graph that is a chain. In this type of graph, $CPGE$ can be solved when there is at most one robot. At the other extreme, $CPGE$ can trivially be solved with n robots in a ring of size n (the omniscient daemon first defines a sense of direction on the ring, and then, repeatedly at each round, directs each robot to move to the next vertex).

While determining the maximal number k of robots for which $CPGE$ can be solved is very easy for some graphs, it is far from being trivial to capture the topological features of a graph that allow determining the value of the upper bound k. To that end, the paper introduces a *reduction procedure* that transforms any graph into a labeled tree (called *mobility tree*) from which the value of k can be determined. This tree captures the paths that have to be traversed by a single robot at a time.

5.1 Preliminary definitions

In the following we consider only undirected connected graphs where any two vertices are connected by at most one edge. A vertex v is a *leaf* of a graph $G = (S, E)$ if there is a single vertex v' such that $(v, v') \in E$. The degree d of a vertex v is the integer $\left|\{v' \mid (v, v') \in E\}\right|$. A *bridge* is an an edge whose deletion disconnects the graph. A graph without bridge is a *bridgeless* graph. A path from a vertex v to a vertex v' is *simple* if no vertex appears on it more than once.

A graph $G' = (S', E')$ is a *subgraph* of a graph $G = (S, E)$ if $S' \subseteq S$ and $E' \subseteq E$. In that case, we also say that $G = (S, E)$ is a *supergraph* of $G' = (S', E')$. A *non-singleton* subgraph contains at least two vertices. The subgraph $G' = (S', E')$ is *induced* by the set of vertices S', if E' contains all the edges of E whose end-points are in S'. As, in the following, all the subgraphs we consider are induced subgraphs we omit the term "induced" to not overload the presentation.

A subgraph G' is *maximal* with respect to a property P if G' satisfies P, while none of its supergraphs satisfies P. So, "bridgeless" and "non-singleton" are properties that a (sub)graph satisfies or does not satisfy.

5.2 From a graph to a tree: the reduction procedure

Definition 2 *(Mobility Tree) Let the* labeled mobility tree *associated with a graph* $G = (S, E)$ *be the labeled tree* $G' = (S', E')$ *derived from* G *through the following reduction procedure:*

1. Initial labeling. *Each vertex* $v \in G$ *is first labeled as follows:*

   - *Label* 0*: if* v *does not belong to a bridgeless subgraph of* G *and its degree is two;*
   - *Label* 1*: if* v *is a leaf of* G *or belongs to a non-singleton bridgeless subgraph of* G*;*
   - *Label* 2*: otherwise.*

2. Compression*. Each maximal non-singleton bridgeless subgraph of G is reduced to a vertex with label* 1*.*

Figure 1 shows an example of the previous reduction procedure. The graph G is depicted in Figure 1(a). The result of the initial labeling of its vertices described in Figure 1(b). The non-singleton maximal bridgeless subgraphs of G are surrounded by a circle in that figure. Finally, the resulting labeled mobility tree obtained from the compression of the non-singleton maximal bridgeless subgraphs is shown in Figure 1(c).
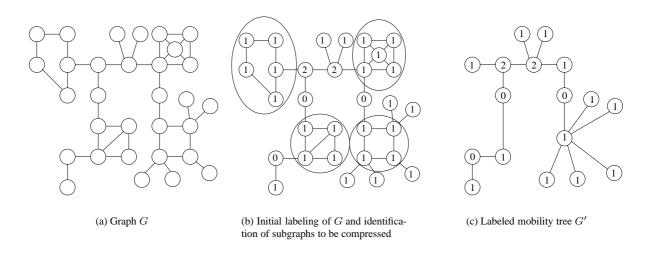

(a) Graph G

(b) Initial labeling of G and identification of subgraphs to be compressed

(c) Labeled mobility tree G'

Figure 1: A graph G and its associated labeled mobility tree G'

The mobility tree of a graph G is intended to point out the noteworthy features of G as far the solvability of $CPGE$ is concerned. First, it points out those subgraphs of G (corresponding to vertices with label 1 in the mobility tree) where $CPGE$ could be solved in each of such subgraphs in isolation with a number of robots equal to the number of vertices of the subgraph. These subgraphs are indeed either leafs of G or non-singleton bridgeless subgraphs of G (where the omniscient deamon can define a sense of direction). Second, the mobility tree shows those paths of G that have to be traversed by a single robot at a time to move from one of the previous subgraphs of G to another one in order to extend the solvability of $CPGE$ in G. Let us therefore introduce the notion of Mutual Exclusion Path.
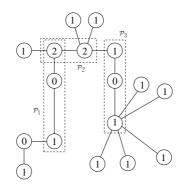

Figure 2: Examples of mutual exclusion paths in a mobility tree

Definition 3 *(Mutual Exclusion Path) Let $\mathcal{P}$ be a path $(v, v^1), (v^1, v^2) \ldots (v^m, v')$ of the mobility tree G' from vertex v to v'. $\mathcal{P}$ is a mutual exclusion path of G' iff:*

- *The labels of v and v' are different from 0;*
- *If there are vertices v^h, $1 \le h \le m$, (i.e., the path from v to v' contains more than one edge), those vertices are labeled 0.*

As an example, Figure 2 shows three mutual exclusion paths, $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}_3$, of the labeled mobility tree shown in Figure 1(c).

Definition 4 *(Length of a Mutual Exclusion path) In a labeled mobility tree $G' = (S', E')$, let the length of a Mutual Exclusion path between any two vertices $v, v' \in G'$ be the number of edges from v to v' augmented with j, where j is the number of vertices with label 2 in that path.*

The length of $\mathcal{P}_1$ depicted in Figure 2 is $2 + j = 3$ (as $j = 1$). The length of $\mathcal{P}_2$ is $1 + j = 3$ (as $j = 2$) while the length of $\mathcal{P}_3$ is $2 + 0 = 2$ (as $j = 0$). Intuitively, *the length of a mutual exclusion path* represents the the minimum number of vertices that have to be initially empty (i.e., without robot assignment) in order the robots be able to solve the $CPGE$ problem *with respect to that path*. Therefore computing the maximal length of the mutual exclusion paths of a mobility three associated with a graph G becomes a key factor to compute the upper bound on the number of robots to keep $CPGE$ solvability in G.

Definition 5 *For any $p > 0$ and $q \geq 0$, let $\mathcal{G}(p, q)$ the set of graphs such that $\forall\, G \in \mathcal{G}(p, q)$ we have:*

- *G has p vertices;*
- *q is the maximal length of the mutual exclusion paths of the mobility tree associated with G.*

Two graphs belong to the same class $\mathcal{G}(p, q)$ if they both have the same number of vertices p and the same maximal length q of the mutual exclusion path of their respective mobility trees. As we are about to see in the next section, the maximal number of robots for which the $CPGE$ problem can be solved in any graph in $\mathcal{G}(p, q)$ is $k = p - q$.

5.3 Upper bound

Theorem 1 *Let G be a graph of class $\mathcal{G}(p, q)$. There exists no algorithm that solves $CPGE$ for G from an arbitrary configuration including more than $k = p - q$ robots.*

Proof The proof is by contradiction. Let us assume that there is an algorithm A that solves $CPGE$ for a graph $G \in \mathcal{G}(p, q)$ and any initial configuration with at least $p - q + 1$ robots. There is no restriction on A: its computational power is the one of a Turing machine with an unbounded memory.

The contradiction for the case $q = 0$ is obvious: it is not possible to place $p + 1$ robots on a graph with p vertices without violating the vertex mutual exclusion property. So, the rest of the proof considers $q > 0$.

Let us observe that, due to the vertex mutual exclusion property, any configuration reachable from the initial configuration, contains $q - 1$ vertices without robots. According to definition of $\mathcal{G}(p, q)$, q is the maximal length of the mutual exclusion paths of the mobility tree of G. Let u be such a path, with X and Y being its end-point vertices. According to the labels of X and Y, three cases can be distinguished (but thanks to the labeled mobility graph abstraction, the reasoning is simple and identical in all cases).

- Both X and Y have label 1. This means that, if the path from X to Y in G contains more than one edge, its vertices different from X and Y have degree 2 and are labeled 0.

  Let G_X and G_Y be the the maximal subgraphs of G that include X and Y, respectively, and satisfy the following property P: any simple path in G from any $v \in G_X$ to any $v' \in G_Y$ includes both X and Y. (See an illustration in Figure 3.) As both X and Y have label 1, it follows from Definition 4 that u is a sequence of $q + 1$ vertices (and, as already observed, each vertex in this sequence different from X and Y has degree 2).

  Let us consider the initial configuration where the $q - 1$ vertices without robots are exactly the vertices of $u \setminus \{X, Y\}$. Then, the $p - (q - 1)$ robots fill completely all the vertices of both subgraphs G_X and G_Y. As there are only $q - 1$ vertices without robots (the vertices of $u \setminus \{X, Y\}$), it follows that no robot in a vertex $v \in G_X$ can move to a vertex $v' \in G_Y$ (or vice-versa) without violating either the vertex mutual exclusion property or the edge mutual exclusion property, which proves the theorem for that case.

- X and Y have label 2 and 1, respectively. We conclude from Definition 4 and the labels of X and Y that the path u is a sequence of q vertices. Let G_X^1, G_X^2 (and possibly G_X^3, G_X^4, ...) denote the maximal subgraphs of G that do not contain X and such that any path in G from any vertex $v \in G_X^1 \cup G_X^2 \cup \cdots$ to any vertex $v' \in G_Y$ (where G_Y includes Y), includes both X and Y (see Figure 4).
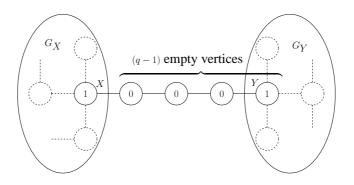
Figure 3: Mutual Exclusion path, from a label 1 to a label 1, of length $q = 4$
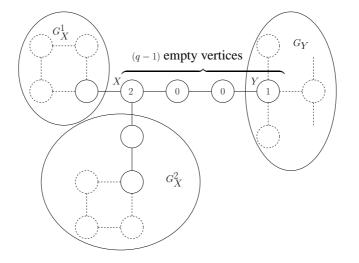

Figure 4: Mutual Exclusion path, from a label 1 to a label 2, of length $q = 4$

Similarly to the previous case, let us consider the configuration where the $q - 1$ vertices without robots are the vertices of $u \setminus \{Y\}$. As previously, the $p - (q - 1)$ robots fill completely G_Y, G_X^1 and G_X^2 (and G_X^3, G_X^4, ... if they exist). As before, the chain from X to Y does not contain enough vertices without robots in order for a robot located at a vertex $v \in G_X^1 \cup G_X^2 \cup G_X^3 \cup \cdots$ to move to a vertex $v' \in G_Y$ (or vice-versa), without violating either the vertex mutual exclusion property or the edge mutual exclusion property. This proves the theorem for the second case.

- Both X and Y have label 2. Due to Definition 4, u is then a sequence of $q - 1$ vertices. Let G_X^1, G_X^2 (and possibly G_X^3, G_X^4, ...), and G_Y^1, G_Y^2 (and possibly G_Y^3, G_Y^4, ...) be the maximal subgraphs of G such that $X \notin G_X^1 \cup G_X^2 \cup \dots$, $Y \notin G_Y^1 \cup G_Y^2 \cup \dots$, and the maximal subgraphs of G such that any path in G from any $v \in G_X^1 \cup G_X^2 \cup \cdots$ to any $v' \in G_Y^1 \cup G_Y^2 \cup \cdots$ includes both X and Y (see Figure 5).

Let us consider the configuration where the $q-1$ vertices without robots are all the vertices of u. The $p-(q-1)$ robots fill completely G_X^1, G_X^2, G_Y^1 and G_Y^2 (and G_X^3, G_Y^3, ... if they exist). It follows that, despite the fact that all the vertices of the path u have no robot, there are not enough vertices without robots to allow a robot in a vertex $v \in G_X^1 \cup G_X^2 \cup \dots$ to move to a vertex $v' \in G_Y^1 \cup G_Y^2 \cup \dots$, which proves the theorem for this last case.
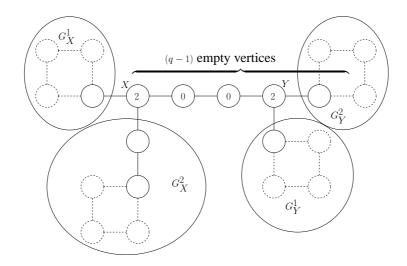
$\square_{Theorem\ 1}$

Figure 5: Mutual Exclusion path, from a label 2 to a label 2, of length $q = 5$

6   Conclusion

This paper has investigated the Constrained Perpetual Graph Exploration problem ($CPGE$). That problem consists in making as much as possible mobile entities (robots) to visit infinitely often each vertex of a connected undirected graph, under the constraint that two robots can neither be simultaneously at the same vertex, nor concurrently travel the same edge. The paper has established an upper bound on the number of robots for which $CPGEn$ can be solved. To make the result as strong as possible, that bound had been proved under the strong assumption that the robots behave synchronously and their movements are coordinated by an omniscient daemon. It has been shown that this upper bound is related to the topological structure of the graph that has to be visited, thereby defining classes of graphs having the same upper bound.

An interesting open problem consists in refining this upper bound when each robot has a bounded memory, or when its current knowledge is restricted to a limited part of the graph (dynamically defined at each round from its current position in the graph).

References

[1] Albers S. and Henzinger M.R., Exploring Unknown Environments. *SIAM Journal on Computing*, 29(4):1164-1188, 2000.

[2] Awerbuch B., Betke M., Rivest R.L. and Singh M., Piecemeal Graph Exploration by a Mobile Robot. *Information and Computation*, 152(2):155-172,1999.

[3] Bender M.A. and Slonim D., The Power of Team Exploration: Two Robots can Learn Unlabeled Directed Graphs. *Proceedings of the 35th Annual IEEE Symposium on Foundations of Computer Science (FOCS'94)*, IEEE Computer Press, pp. 75-85, 1974.

[4] Budach L., Automata and Labyrinth. *Mathematische Nachrichten 1948-1999*, 86(1):195-282, 1978.

[5] Dobrev S., Jansson J., Sadakane K. and Sung W.K., Finding Short Right-Hand-on-the-Wall Walks in Undirected Graphs. *Proc. of the 12th Colloquium on Structural Information and Communication Complexity (SIROCCO'05)*, Springer-Verlag, LNCS #3499, pp. 127-139, 2005.

[6] Flocchini P., Ilcinkas D., Pelc A. and and Santoro N., Computing Without Communicating: Ring Exploration by Asynchronous Oblivious Robots. *In Proc. 11th Int'l Conference On Principles Of Distributed Systems (OPODIS'07)*, Springer-Verlag, LNCS #4878, pp. 105-118, 2007.

[7] Fraigniaud P., Ilcinkas D., Peer G., Pelc A. and Peleg D.. Graph Exploration by a Finite Automaton. *Theoretical Computer Science*, 345(2-3):331-344, 2005.

[8] Fraigniaud P., Ilcinkas D., Rajsbaum S. and Tixeuil S., Space Lower Bounds for Graph Exploration via Reduced Automata. *In Proc. 12th Colloquium on Structural Information and Communication Complexity (SIROCCO'05)*, Springer-Verlag, LNCS #3499, pp. 140-154, 2005.

[9] Gasieniec L., Klasing R., Martin R.A. Navarra A. and Zhang X., Fast Periodic Graph Exploration with Constant Memory. *In Proc. 14th Colloquium on Structural Information and Communication Complexity (SIROCCO 2007)*, Springer-Verlag, LNCS #4474, pp. 26-40, 2007.

[10] Grossi R., Pietracaprina A. and Pucci G., Optimal Deterministic Protocols for Mobile Robots on a Grid. *Information and Computation*, 173(2):132-142, 2002.

[11] Ilcinkas D., Setting Port Numbers for Fast Graph Exploration. *In Proc. 13th Colloquium on Structural Information and Communication Complexity (SIROCCO'06)*, Springer-Verlag, LNCS #4056, pp. 59-69, 2006.

[12] Panaite P. and Pelc A., Impact of Topographic Information on Graph Exploration Efficiency. *Networks*, 36(2):96-103, 2000.

[13] Rollik H.A., Automaten in Planaren Graphen. *Acta Informatica*, 13:287-298, 1980.

[14] Suzuki I. and Yamashita M., Distributed Anonymous Mobile Robots: Formation of Geometric Patterns. *SIAM Journal on Computing*, 28(4):1347-1363, 1999.

[15] Yared R., Defago X. and Wiesmann M., Collision Prevention Using Group Communication for Asynchronous Cooperative Mobile Robots. *In Proc. 21st Int'l IEEE Conference on Advanced Information Networking and Applications (AINA 2007)*, IEEE Computer Press, pp. 244-249, 2007.