Broadcasting with Mobile Agents in Dynamic Networks

Shantanu Das
Aix-Marseille Université, CNRS, Université de Toulon, LIS, Marseille, France
shantanu.das@lis-lab.fr

Nikos Giachoudis
DCSBI, University of Thessaly, Lamia, Greece
ngiachou@gmail.com

Flaminia L. Luccio
DAIS, Università Ca' Foscari Venezia, Italy
luccio@unive.it

Euripides Markou
DCSBI, University of Thessaly, Lamia, Greece
emarkou@dib.uth.gr

Abstract

We study the standard communication problem of broadcast for mobile agents moving in a network. The agents move autonomously in the network and can communicate with other agents only when they meet at a node. In this model, broadcast is a communication primitive for information transfer from one agent, the source, to all other agents. Previous studies of this problem were restricted to static networks while, in this paper, we consider the problem in dynamic networks modelled as an evolving graph. The dynamicity of the graph is unknown to the agents; in each round an adversary selects which edges of the graph are available, and an agent can choose to traverse one of the available edges adjacent to its current location. The only restriction on the adversary is that the subgraph of available edges in each round must span all nodes; in other words the evolving graph is constantly connected. The agents have global visibility allowing them to see the location of other agents in the graph and move accordingly. Depending on the topology of the underlying graph, we determine how many agents are necessary and sufficient to solve the broadcast problem in dynamic networks. While two agents plus the source are sufficient for ring networks, much larger teams of agents are necessary for denser graphs such as grid graphs and hypercubes, and finally for complete graphs of n nodes at least $n-2$ agents plus the source are necessary and sufficient. We show lower bounds on the number of agents and provide some algorithms for solving broadcast using the minimum number of agents, for various topologies.

2012 ACM Subject Classification Networks → Network algorithms; Theory of computation → Distributed algorithms; Theory of computation → Graph algorithms analysis

Keywords and phrases Distributed Algorithm, Dynamic Networks, Mobile Agents, Broadcast, Constantly Connected, Global visibility

Digital Object Identifier 10.4230/LIPIcs.OPODIS.2020.24

Funding Partially supported by DAIS - Ca' Foscari University of Venice (IRIDE program). Most of this work was done when Shantanu Das and Euripides Markou visited Ca' Foscari University.

1 Introduction

We are interested in communication problems for mobile agents moving in a network. The classical problems of broadcast or convergecast deal with the dissemination of information in the network. In the case of message passing networks, broadcast is achieved by spreading the information from the source node to all other nodes. For a system of mobile agents, the equivalent problem is the propagation of information from one source agent to all other agents in the system. Such problems are relevant for teams of mobile sensor robots sent on data collection missions. We assume that the agents autonomously move along the edges of a graph that represents the network; when two agents are at the same node, they can communicate and share information. We would like to stress here that the agents are not allowed to use any means of communicating at a distance (e.g., due to security reasons). The information to be broadcast can be transferred only when agents meet physically.

The problem of broadcasting has been originally investigated in message passing multi-hop radio networks (see e.g. [33, 19]). Previous studies on broadcast and other communication problems have focussed on the efficiency of performing the task, either in terms of time taken [6], or in terms of energy expended [9]. A slightly different line of research considers the problem of broadcast in the presence of faults and the objective is to tolerate as many faults as possible. The faults can be missing links or nodes [20] in the network or loss of messages [21], in case of message passing networks. Recently there has been a lot of interest in so called *dynamic networks* which model both faults and changes in network topology in a uniform manner by considering that the network may change in each round during the execution of the algorithm. The *evolving graph* model [17] represents a dynamic network by a sequence of graphs $\mathcal{G} = G_1, G_2, \dots$ based on the same set of nodes V but the set of edges changes in each round i, i.e., each graph $G_i = (V, E_i)$ is a spanning subgraph of the underlying graph $G = (V, \cup E_i)$, which is called the footprint of the dynamic network. For solving most problems, some assumptions about the connectivity of the dynamic network need to be made. In this paper, we consider the model of *constantly connected* dynamic networks where in round i, the graph G_i is assumed to be connected. No other assumptions are made about the network. This means that in some cases, the graphs G_i and G_{i+1} in two consecutive rounds may differ completely in the set of available edges. To show correctness of our algorithms, we will assume that an adversary having knowledge of the algorithm, chooses the graph G_i in each round (respecting the connectivity constraint). This assumption is based on a worst case scenario and similar to the model of T-interval connected networks studied previously [26, 27], where the network is assumed to contain a stable spanning tree for a continuous period of T rounds (with $T = 1$ in our case).

When the underlying graph G is sparse, there are fewer edges so the adversary has less choice about possible changes in the network, while if G is a dense graph, the adversary has more choice about which subset of edges to make available, and it can drastically change the network which makes it difficult to design algorithms for these cases. We will consider different network topologies, including sparse topologies, e.g., when G is a ring or a cactus, as well as denser topologies, e.g., when G is a grid, a hypercube or a complete graph. For each topology we will design algorithms to solve the broadcast problem under the assumption that agents have global visibility allowing them to see the entire graph and the location of other agents in any round of the algorithm. We denote by k, the number of agents, other than the source that participate in the algorithm. We will show that the problem becomes easier when there are more agents since at least one of them may be able to reach the source agents and thus make progress in the propagation of the message. In fact we show tight lower bounds on the value of k necessary to solve the broadcast problem in various topologies. We formally define the problem as follows:

▶ **Problem 1** (The broadcast problem). *Given a constantly connected dynamic network $\mathcal{G}$ based on an underlying graph G consisting of $n \geq 2$ nodes, a source agent that has a message $\mathcal{M}$ and $k \geq 1$ other agents that are initially located at distinct nodes of the network, the goal is to broadcast this message $\mathcal{M}$ to all the agents.*

Related Work

Dynamic networks using the evolving graph model or the equivalent notion of time-varying graph model have been studied in [5, 23, 28, 30]. Earlier studies have been devoted to message passing networks in a dynamic setting, often assuming that the future changes to the network are known a priori to the algorithm designer, or, that the dynamicity follows a predefined pattern of repetition as, e.g., in [4]. In message passing dynamic networks, the problem of information dissemination has been studied in [2, 4, 7, 27, 32]. The investigation of mobile agents on dynamic networks started only recently. When agents know the dynamicity of the graph, the problem of exploring a graph in the fastest possible way has been studied in several papers, e.g,. [16, 31]. In the case of constantly connected dynamic graphs, the exploration problem has been solved by Ilcinkas et al. [26, 24] in $\mathcal{O}(n)$ rounds for rings and in $2^{\mathcal{O}(\sqrt{n})}n$ rounds for cactus graphs.

The scenario when agents do not know the dynamicity of the graph as in this paper, has been mostly studied under restrictive assumptions about the dynamicity, such as periodic [18, 25] or recurrent [26] graphs; while another line of research has looked at probabilistic dynamicity [34]. In the adversarial model (also called the *unknown adversary* model), the adversary chooses the dynamicity of the network, and the agents have no prior knowledge about it. This scenario is the most challenging for designing algorithms and most prior work under this model has been for very simple topologies, namely rings and tori. In this setting, the problem of exploration with termination has been studied for constantly connected rings [14], while the exploration of dynamic tori has been studied in [22] with the assumption that each ring in the torus is constantly connected. The problem of gathering many agents at a node [15] or periodically patrolling the nodes using many agents [13] has been studied for constantly connected dynamic ring networks (See the recent survey by Di Luna [29] for many of the above results). The only previous work in this model which considers arbitrary topologies is by Balev et al. [3] who studied the problem of *cops and robbers* on sparse graphs of arbitrary topology. In this problem, a team of agents called cops have to capture (i.e., meet) a malicious agent called the robber, while the cops and the robber move in alternate rounds (the adversary may change the graph after both have moved). The similarity of this problem with the broadcast problem is only the first step when one of the ignorant agents needs to meet the source agent (although they can collaborate unlike in the cops and robbers problem). One major difference between the problems is that the cops can choose their location in the graph, unlike the agents in our problem that are placed by the adversary. Consequently the approach used in [3] cannot be adapted to our problem. Moreover the bounds on the team size even for simple topologies are different for the two problems. Only in the trivial case of tree networks (when the graph is essentially static) both problems allow a solution with a single ignorant agent (cop).

In static networks, the communication problems of broadcast among mobile agents were studied by Anaya et al. [1] for agents with limited energy, while Czyzowicz et al. [8, 9] studied the problem with the objective of energy optimization. Other types of faults that have been considered for static networks are faulty agents in the context of collaborative patrolling [10] or the presence of a mobile adversary that blocks the path of agents [11, 12].

Our Contributions

In this paper we consider the broadcast problem for mobile agents in dynamic networks with various underlying topologies. For each topology, we determine the minimum number of agents that makes the problem solvable, which apparently depends on the density of the underlying graph G or, the number of redundant edges in G. For tree networks, the adversary can never delete an edge without disconnecting the graph, thus broadcast is always solvable for any number of agents. When the underlying graph is a ring, we show that at least 2 agents (apart from the source agent), are needed, except for small rings of less than 5 nodes. For cactus graphs, which are collections of rings that can pairwise intersect in no more than one node, we show that the number of agents necessary must be more than the number of cycles. We then consider denser and regular graphs. For grid graphs, where the total number of edges in G is still linear in n, we show that agent teams of size $\Omega(n)$ are needed for broadcast. For the special case of grids with only two rows, we have tight results and a strategy for solving broadcast using one more than the minimum number of agents. For complete graphs, which are the densest graphs we show a tight result that $k = n - 2$ agents are necessary and sufficient. Finally we consider hypercube networks, and we show that almost half of the nodes of the graph must be occupied to succeed in solving broadcast. For the special case of 3-dimensional hypercube, we show a tight result of $k = n/2$ agents, while for higher dimensional hypercubes, there is a gap between the lower and upper bounds on the team size needed for solving broadcast.

The paper is organized as follows: In Section 2 we introduce the model and the required background, followed by some preliminary observations. In Section 3 we present solutions for sparse network topologies such as rings and cactus graphs. Section 4 considers grids graphs, while in Section 5 we present solutions for dense networks including hypercubes and complete graphs. To the best of our knowledge, these are the first results on broadcast with mobile agents in dynamic graphs, and unlike previous work in this model, we consider various distinct topologies, providing new techniques for dealing with the dynamicity in these networks.

2 The Model

2.1 The network and the agents

The network topology is given by a connected graph $G = (V, E)$ with $n = |V|$ nodes. The network is locally oriented in the following sense. All edges incident to a node have distinct port labels. However, the network is dynamic – not all edges are available at all times. We model the network as *Constantly connected* Dynamic Graphs denoted by $\mathcal{G} = \{G_0 = (V, E_0), G_1 = (V, E_1), \ldots\}$, as a sequence of static graphs, where G_r corresponds to the graph at round r. We emphasize that apart from the restriction that the graph remains connected at any time, there is no other assumption or restriction with respect to which or how many links might fail at a time. For instance, an adversary might keep deactivated any number of links forever, as long as the graph remains connected. Thus, for any $r > 0$, G_r is any connected spanning subgraph of the original graph G. The distance between nodes u and v in G is denoted by $d_G(u, v)$.

The agents: The agents are autonomous entities with distinct identifiers. Each agent has its own internal memory and is able to move along the available edges of the graph. The agents cannot leave marks on the nodes or the edges of the graph. The agents are initially located at distinct nodes of the network, they all start in the same initial state, and they execute the same deterministic algorithm. They have *global visibility* and they move in *synchronous* steps, i.e., time is discretized into atomic time units called rounds. During each round r, each agent can see the graph G_r and the location of all agents in G_r along with their identifiers, and can distinguish which agents have the message $\mathcal{M}$ (called *source agents*), and which agents do not (called *ignorant agents*). Based on this information, the agent can decide to stay at the current node or move to a neighboring node in G_r. In the latter case, the agent arrives at its destination node at the end of the round. At the start of the next round $r+1$, the adversary chooses the graph G_{r+1}, and the agents execute the next step of the algorithm. In the initial round, there is exactly one source agent and k ignorant agents in the network.

The adversary: The adversary can decide the initial placement of all the agents in the network, and in each round the adversary chooses the graph G_r which represents the available links in the network for that round. The adversary may have knowledge of the algorithm and can use this knowledge for deciding the placement of agents and the dynamicity of the network (subject to the connectivity constraint as described).

Unknown Adversary: The agents do not have any knowledge of the adversary, and thus they do not know the dynamic network $\mathcal{G}$ in advance.

As mentioned before, the adversary in this model is quite powerful, which makes it necessary to make some strong assumptions about the capabilities of the agents. In particular the global visibility makes it easier to coordinate among the agents as they all have the same knowledge about the network in each round. The assumption about distinct initial locations is not strictly necessary, since the agents could move to distinct nodes by virtue of their distinct identifiers. On the hand, if the agent do not have distinct identities but start from distinct locations, it is possible to assign distinct identifiers to the agents at the start of the computation, due to the global visibility assumption and the fact that there is a uniquely identifiable source agent initially. For simplifying the discussion, we assume agents have distinct identities and start at distinct locations. Moreover, we shall describe the algorithms in a centralized manner, describing which agents performs which operations in any round. It is evident that the agents executing the same algorithm, can autonomously decide their role in the computation.

2.2 Preliminaries

We make some preliminary observations about the problem.

▶ **Observation 1.** *Given a constantly connected dynamic network $\mathcal{G}$ based on the underlying graph G consisting of n nodes, $k \geq n-2$ ignorant agents starting from distinct nodes of the network, can solve the broadcast problem.*

Proof. Since there are at least $n-2$ ignorant agents on distinct nodes and one node is occupied by the source agent, there is at most one empty node. Thus, in any connected graph G_i there would be a path of length at most two between a source agent and an ignorant agent. These two agents would meet in this round. So, we reduced the number of ignorant agents. The agents can now spread to distinct nodes by virtue of their distinct identities and we can repeat the same argument. ◀

The above result provides a general upper bound on the team size needed for solving broadcast. We will present smaller lower bounds for specific topologies. For the special case of trees, the adversary cannot block any edge with losing connectivity. Hence we have the following trivial upper bound for trees.

▶ **Observation 2.** *If G is a tree then broadcast can be solved for any $k \geq 1$.*

Proof. The adversary cannot block any edge without disconnecting the graph. Thus, the graph is static and in each step each agent can move one step closer to the node containing the source agent, thus in $O(D)$ time all agents would be colocated with the source agent and we solve broadcast in $O(D)$ time, where D is the distance of the farthest agent from the source. ◀

3 Broadcast in sparse graphs

In this section, we will study the broadcast problem with agents in sparse graphs. The simplest non-trivial network is the ring topology.

▶ **Theorem 3.** *If G is a ring of size $n \geq 5$, then broadcast can be solved if and only if $k \geq 2$ If G is a ring of size $n < 5$, then broadcast can be solved for any $k \geq 1$.*

Proof. Consider a ring of size $n \geq 5$, with one source that has the information $\mathcal{M}$ and one ignorant agent. In each round, the adversary can remove an edge on the shortest path between the two agents. Note that, the longer path is always of size at least 3, thus the agents cannot meet on this path in this round. Hence the two agents can never meet.

Now we show that if there are at least two ignorant agents ($k \geq 2$), then broadcast is possible. The two agents can try to reach the source agent by moving towards it from opposite directions, then at each step one of the agents gets closer, and eventually one of the agents would reach the source and obtain $\mathcal{M}$. At this stage there are 2 source agents, they can traverse the ring in opposite directions, thus at least one of them will soon meet the remaining ignorant agent and broadcast is solved.

The impossibility of solving broadcast with $k = 1$, does not hold for rings of size 3 as in this case any path between the source and the other agent is of size at most 2, and since one of these paths must be available, the two agents can meet in one step and solve the problem. For rings of size 4, if the longer path between the source agent and an ignorant agent has length 3, then one of the agents can always move such that both paths between the agents are of length 2. Then within the next step, the two agents meet in one of those paths, similarly as in the case of rings of size 3. Thus, broadcast is solvable for rings of size $n < 5$ with any $k \geq 1$. ◀

We now make the following observation that will allow us to generalize the results from rings to other graphs containing cycles.

▶ **Lemma 4.** *If G contains a cycle C of length at least 3, such that there is a single node $v \in C$ that is connected to nodes in $G \setminus C$, then the adversary can always prevent at least one agent located in $C \setminus v$ from reaching node v, thus trapping the agent in cycle C.*

Proof. Consider an agent located at a node $u \in C \setminus v$. Since the cycle must be of length at least 3, the longer path from node u to node v is of length ≥ 2. If the adversary always blocks the shorter path from the agent's location to node v, then the agent cannot reach node v. The only way to get out of the cycle is passing through node v, so the agent is forever trapped in C. ◀

▶ **Lemma 5.** *If G is a ring of size $n \geq 3$, given any node $v \in G$, if there are two agents at distinct nodes of G, then there is an algorithm to ensure that within n rounds either (i) the two agents meet at a node of G or (ii) at least one of the agents (chosen by the algorithm) can reach node v.*

Proof. Let us call the two agents A and B. If we require agent A to reach node v, then the algorithm asks agent B to move along the path containing agent A and then node v in this order (if agent B was already at node v then we first move it to any neighbouring node of v). In each round, only one edge of G may be unavailable, so either agent B will move closer to agent A or agent A will move closer to node v. So eventually either condition (i) or (ii) will be true. ◀

In the following we consider cactus graphs which can be seen as combinations of trees and rings.

▶ **Definition 6.** *A* cactus graph *is a connected graph in which any two simple cycles have at most one node in common.*

In cactus graphs, the size of the team depends on the number and sizes of the cycles of the graph, as follows:

▶ **Lemma 7.** *If G is a cactus graph of size n having $c_1 \geq 1$ cycles of length < 5 and no larger cycles, then broadcast can be solved if and only if $k \geq c_1$.*

Proof. Consider the family of cactus graphs obtained from a line of length c_1 by attaching to each node of the line a cycle of length < 5. Each cycle thus satisfies the conditions of Lemma 4.

If $k < c_1$, the total number of agents is $k + 1 \leq c_1$, so the adversary can place each agent in a distinct cycle, including the source agent. No agent can leave its cycle due to Lemma 4. So, no two agents can meet, thus broadcast is not possible.

For $k \geq c_1$, broadcast is solvable in any cactus graph with c_1 small cycles. To prove this it is enough to analyze the case where at least one cycle has at least two agents, either two ignorant ones or one ignorant agent and a source. This is because if an agent is outside of any cycle it can always move to a cycle (all non cyclic edges are available in each round). If two agents are in a cycle and none of them is the source, then one of the agents can leave the cycle within at most 2 steps (both agents try to move towards an elected exit by approaching it from different directions). The agent that leaves the cycle can move towards the source, until it reaches another cycle.

Thus, one agent will eventually reach the cycle containing the source. This agent can meet the source and obtain the information $\mathcal{M}$ due to Theorem 3. Now, there are two source agents in the same cycle, and thus, one of them can leave the cycle as described before within at most 3 steps. This source agent reaches another ring containing an ignorant agent, the information is propagated and we have again two source agents in a cycle. Repeating the same algorithm, all agents will eventually learn the information, and thus we can solve broadcast. ◀

▶ **Lemma 8.** *If G is a cactus graph of size n having c_2 cycles of length ≥ 5 and no cycles of smaller length, then broadcast can be solved if and only if $k \geq c_2 + 1$.*

Proof. Consider the family of cactus graphs obtained from a line of length c_2 by attaching to each node of the line a cycle of length ≥ 5.

Suppose that $k < c_2 + 1$. Then, the adversary places each of the $k \leq c_2$ agents in a distinct cycle and the source in one of these cycles. Due to Theorem 3, the source and the other agent cannot meet, since the ring is of length ≥ 5. At the same time, due to Lemma 4 the adversary can trap each other agent in its cycle. In the cycle that contains the source, at most one of the two agents can exit this cycle. Even if the source agent exits this cycle and enters another cycle the configuration is similar to the initial one. Hence, no two agents can ever meet and therefore the problem is unsolvable.

To prove that $k \geq c_2 + 1$ agents are enough to solve broadcast, we first show that at least one ignorant agent can reach the source and can become a new source. As in the proof of Lemma 7, we assume all agents move to some cycle if they are not in a cycle. If the source agent is in the same cycle with at least two ignorant agents, by Theorem 3 both these agents can become sources. If not, then, given $k \geq c_2 + 1$, there must be some cycle with two or more ignorant agents and all except one of these agents can leave the cycle to reach another cycle. Eventually two or more ignorant agents would reach the same cycle as the source, and again applying Theorem 3, all these agents would become source agents. Thus we have now at least $x \geq 3$ source agents in a cycle. Furthermore each of the remaining $k - x + 1$ ignorant agents are alone in some cycle. This implies that the number of empty cycles (cycles without any agent) are at most $x - 3$. Among the x source agents, $x - 1$ of them can move to another cycle. Whenever these source agents arrive at an empty cycle, at most one of them may be trapped. In total $x - 3$ source agents can be trapped, thus at least two source agents can reach any other cycle that contains an ignorant agent, so this agent will meet a source. Hence, all ignorant agents will eventually become sources and thus broadcast can be solved. ◀

▶ **Theorem 9.** *Let G be a cactus graph of size n having c_1 cycles of length < 5 and c_2 cycles of length ≥ 5, then:*
- *If $c_2 = 0$, broadcast can be solved if and only if $k \geq c_1$.*
- *If $c_2 > 0$, broadcast can be solved if and only if $k \geq c_1 + c_2 + 1$.*

Proof. If $c_2 = 0$, then the cactus graph has only c_1 cycles of length < 5, and in view of Lemma 7 broadcast can be solved if and only if $k \geq c_1$.

On the other hand if $c_2 > 0$ and $c_1 = 0$, then in view of Lemma 8 broadcast can be solved if and only if $k \geq c_2 + 1$, and thus the second condition holds.

Finally, if $c_1 > 0$ and $c_2 > 0$, similarly as in the proofs of Lemmas 7 and 8, we can construct a cactus graph from a line of length $c_1 + c_2$ by attaching a cycle of length < 5 to each of the first c_1 nodes and attaching a cycle of length ≥ 5 to each of the remaining c_2 nodes. Now, if $k \leq c_1 + c_2$, then the adversary places each ignorant agent in a distinct cycle, and places the source agent in the last big cycle C of length ≥ 5. Since there is at most one ignorant agent and the source in cycle C of length ≥ 5, they cannot meet (see the proof of Theorem 3). Furthermore no other agent (in a cycle different than C) can leave its cycle due to Lemma 4. The source agent may escape from the cycle C and reach another cycle C'. If the cycle C' is big (size ≥ 5), then as before the source agent would not be able to meet the only agent that is in cycle C'. On the other hand, if the source reaches a small cycle (size < 5) it may meet the ignorant agent in that cycle, so we will have two sources; however at most one of the two can leave this cycle. Thus the agents in the big cycles would never meet any source agent. Thus broadcast can not be solved.

We now show how to solve broadcast using $k \geq c_1 + c_2 + 1$ ignorant agents. First, as argued before, any agent that is not on a cycle can move to the nearest cycle. Since there are more agents than cycles, there are some cycles that contain multiple agents. In any such cycle, one of the agents can move to a neighboring empty cycle if there is one. Repeating this process, we can distribute the agents such that there is at least one agent in each cycle.

Let C be the cycle that contains the source. In any cycle other than C, if there are more than one ignorant agents, all except one of them can move to another cycle that is closer to cycle C. Repeating this process, we will reach a configuration where there will be at least 2 ignorant agents and the source in cycle C and exactly one agent in each other cycle. Now it is easy to solve broadcast from this configuration. Using the ring algorithm (Theorem 3) all ignorant agents in cycle C would become sources. Since we have at least three source agents now, at least two of them can move to a different cycle. In any other cycle reached by those two source agents, there is one ignorant agent, so we can apply the same algorithm and have 3 source agents in this cycle. Repeating this process all ignorant agents will become sources and broadcast is solved. ◀

4 Broadcast in Grids

We now study grid graphs which are slightly more dense than rings or cactuses. Even for 2-dimensional grids, we show that we need $\Omega(n)$ agents to solve broadcast. Let us first consider the simplest grid graph with only two rows (so called ladder graph).

▶ **Theorem 10.** *If G is $2 \times L$ grid graph, then broadcast is unsolvable for $k < L - 1$.*

Proof. Suppose that $k < L - 1$. The adversary can put all k agents in one row consisting of L nodes and the source agent in the other row. So, there is at least one column where both nodes are empty. The adversary would allow this edge and remove all other edges connecting the two rows. So the agents could only move within their respective rows. After some agents move, there would again be some column containing only empty nodes. So the above argument can be repeated. Thus, no agent can leave its respective row at any step, and hence no agent can meet the source. ◀

The above lower bound is almost tight as we can show an upper bound of $k \geq L$ for broadcast in any $2 \times L$ grid graph.

▶ **Theorem 11.** *If G is a $2 \times L$ grid graph, then broadcast is solvable for $k \geq L$.*

The proof is omitted, but we present some basic ideas here. The algorithm for broadcast in $2 \times L$ grids is based on the following lemma which ensures that the ignorant agents can get closer and closer to the source until one of the agents meets the source.

▶ **Lemma 12.** *If G is $2 \times L$ grid graph containing a source agent at origin $[0, 0]$, such that the number of nodes occupied by ignorant agents is strictly greater than the number of unoccupied nodes in G, then there exists a move of a subset of the agents which maintains the ignorant agents in distinct locations and, either (i) one ignorant agent meets the source agent, or (ii) the sum of distances from the ignorant agents to the source agent in G, decreases by at least one.*

To see why the lemma holds, consider any agent x and the path from this agent to the source during the current round of the algorithm (there always exists such a path as the dynamic graph is constantly connected). If all the agents on this path simultaneously move by one edge along this path, and if there are more agents than unoccupied nodes, then we can show more than half of the agents would get closer[1] to the source. Thus the sum of distances from the agents to the source will decrease monotonically and the algorithm makes progress. Note that there are $k \geq L$ agents in distinct locations, there are at most $L - 1$ unoccupied nodes. Thus, there is always some path with more occupied nodes than unoccupied nodes. However the algorithm should always maintain the agents in distinct locations. There are several other technical difficulties that need to be overcome to make the algorithm work, for example, when the source is not in the corner node but somewhere in the interior then we need to partition the grid and apply the lemma selectively on one partition and so on. The resulting algorithm is quite involved and the above approach does not generalize to general grids with more than two rows.

Indeed in larger grids we need a larger team of agents to solve the problem. We will now consider the general case of a $W \times L$ grid graph for $W, L > 2$ and show some lower bounds for broadcast in such grids. We first prove the following technical lemma.

[1] closer in terms of distances in the original graph G

▶ **Lemma 13.** *If G is an $h \times L$ grid graph, with $h \geq 1, L > 2$, then starting from a configuration with $L-1$ agents in each row, the adversary can ensure that there are never more than $L-1$ agents in the bottom row. Moreover, if we add an additional agent at the bottom row, again the adversary can ensure that there are never more than L agents in the bottom row.*

Proof. If $h = 1$ then there is only one row, so the number of agents on the bottom row never changes and the lemma holds trivially. We now prove the lemma by induction on the number of rows. Suppose that the lemma holds for a grid G with $h \geq 1$ rows. We can construct a grid G' by adding an additional row at the bottom with L nodes and $L-1$ agents on distinct nodes. Since the bottom row of G has at most $L-1$ agents (by induction hypothesis), there exists an empty node v in this row. In the grid G' the adversary makes available the edge from v to the node directly below (on the additional row); all other edges between G and the additional row are unavailable. In the current round, no agent can enter the bottom row of G'; although an agent from the bottom row may go up and reach grid G. In the next round if there are $L-2$ agents in the bottom row, and L agents in the row above then the adversary makes available the edge between a node (containing at most one agent) in this row to the node below in the bottom row (and disables all other edges between these two rows). So, either an agent moves to the bottom row or the number of agents in the bottom row remains the same. In the first case, we are back in the initial situation and we could use the same arguments as before. In the second case, the number of agents at the bottom row is smaller than what was initially. Thus the lemma holds for grids of $h+1$ rows and thus by induction for all grids satisfying the conditions of the lemma. Furthermore, note that all arguments remain the same if initially there are $x > L-1$ agents in the bottom row, i.e., the number of agents in the bottom row is never more than x if the higher rows contain at most $L-1$ agents initially. ◀

▶ **Lemma 14.** *If G is a $W \times L$ grid graph, with $W > 2$ then broadcast in unsolvable for $k < (L-1)(W-1)$.*

Proof. We can construct a grid of size $W \times L$ by joining two grids: a grid G^1 of $(h = W-2) \times L$ with a grid G^2 of size $2 \times L$ (by adding L edges between the bottom row of G^1 and the top row of G^2). In grid G^1, we place $L-1$ ignorant agents on each row, at distinct locations, while in grid G^2 we place $k_2 < (L-1)$ agents plus the source agent, as in Theorem 10. Thus the total number of agents is $k = (W-2) * (L-1) + k_2 < (L-1)(W-1)$. We now show that broadcast is not solvable in this graph.

First, if no additional agents enter grid G^2 and the source never leaves G^2 then by Theorem 10, broadcast is not possible as no agent would meet the source in G^2. Furthermore, if the source is in the bottom row, it can never leave this row and thus it cannot leave G^2. In the grid G^1 there are $L-1$ agents in the bottom row and by the Lemma 13 the number of agents on this row does not increase (considering only agents in G^1), so there is at least one empty node on this row. The edge between this node and the grid G^2 is made available and all other edges between the two grids are unavailable. In this case, no agents from G^1 can enter G^2. However, some ignorant agents from G^2 can enter G^1. If x ignorant agents from G^2 enter the grid G^1, then by Lemma 13, the number of agents on the bottom row of G^1 is at most $L-1+x$. In each round, if there is some empty node v on this row, the edge between v and G^2 is the only edge between the two grids that is made available in that round (in this case, no agents can move from G^1 to G^2). Otherwise any node in the bottom row of G^1 can have at most x agents in this round; so, if one edge is available between the two grids, at most x agents can move from G^1 to G^2. Thus, after each round, the number of ignorant agents in G^2 is less than $L-1$ and thus broadcast is not possible by Theorem 10. ◀

We can generalize the above to higher dimensional grids as follows:

▶ **Theorem 15.** *If G is a $(d+1)$-dimensional grid graph of size $W_1 \times W_2 \times \ldots W_d \times L$ where $W_i \geq L \geq 2$ then broadcast in unsolvable for $k < (L-1)(W_1 \cdot W_2 \ldots W_d - 1)$.*

Proof. Consider the subgraph of the grid that is a 2-dimensional grid of size $(W_1 \cdot W_2 \ldots W_d) \times L$. In each round the adversary chooses the available graph to be a connected subgraph of this $2D$ grid, then we can apply Lemma 14 to obtain the above lower bound. ◀

5 Broadcast in Dense graphs

In dense graphs there are many disjoint paths between two nodes and thus there are many possible ways for the adversary to change the network while keeping it connected. In other words, the dynamicity of a (constantly connected) dynamic graph whose footprint is a dense graph, is higher than that of sparser graphs that we studied before. The worst case is when the underlying graph is a complete graph.

5.1 Broadcast in Complete graphs

▶ **Theorem 16.** *If G is a complete graph of size n then broadcast can be solved if and only if $k \geq n-2$ within $O(n)$ steps.*

Proof. If $k < n-2$ then at most $n-2$ nodes are occupied (including the source), and therefore there are at least two empty nodes. The adversary will make available the spanning tree where the source is connected to one empty node and all other occupied nodes are connected to the other empty node. The two empty nodes are connected with an edge. This is a spanning tree of G and in this tree, the distance from the source to any occupied node is more than two. So no agent can meet the source in one step. After one step, during which some agents may move, there will still be at least two empty nodes; thus the same argument can be repeated for any step. Hence broadcast is impossible for $k < n-2$.

If $k \geq n-2$, then by the Observation 1, it is possible to solve broadcast in any arbitrary topology, and thus in a complete graph. ◀

The impossibility result above can be generalized to arbitrary graphs G having the following property.

▶ **Lemma 17.** *Consider a graph G and an integer $k \geq 1$. Suppose that for every possible placement of the source agent and k agents on distinct nodes of G, there always exists a spanning tree of G where the distance from each agent to the source is ≥ 3. Then, broadcast is impossible in G with k ignorant agents.*

5.2 Broadcast in Hypercubes

We now study the problem in hypercube networks as defined below.

▶ **Definition 18.** *A d-*dimensional hypercube *is a graph $H_d = (V, E)$ with $n = 2^d$ nodes labelled with distinct d-bit strings. A node $v_i \in V$, $0 \leq i \leq 2^d - 1$, is connected to the d nodes whose labels differ in exactly one bit from its own. Hereafter, we freely identify nodes with their labels.*

A hypercube H_d consists of two $d-1$ dimensional hypercubes labelled as $[0 * * \cdots *]$ and $[1 * * \cdots *]$, the corresponding nodes of these two hypercubes are connected by edges of dimension d.

▶ **Theorem 19.** *Given a hypercube H_d of dimension $d > 2$, at least $k = n/2 - 1$ ignorant agents are necessary to solve broadcast in the dynamic graph based on H_d.*

Proof. Assume $k < n/2 - 1$. Suppose the source agent is at the node $[00\dots0]$ of H_d; the adversary places all the ignorant agents among the nodes of the sub-hypercube H_{d-1} labelled $[1**\cdots*]$. Out of the $n/2$ nodes in $H_{d-1}[1**\cdots*]$, there must be at least 2 empty nodes. At most one of these two nodes can be a neighbor of the source node $[00\dots0]$ in the other sub-hypercube. So, the other empty node v must be a neighbor of an empty node u in $H_{d-1}[0**\cdots*]$. The adversary chooses the available graph G_i as the union of a spanning tree of $H_{d-1}[0**\cdots*]$ and a spanning tree of $H_{d-1}[1**\cdots*]$, plus the edge (u, v). All other edges of dimension d are missing in G_i. After the agents move in this round, the ignorant agents would still be in sub-hypercube $H_{d-1}[1**\cdots*]$ and the source would be in the other sub-hypercube $H_{d-1}[0**\cdots*]$. Thus, using the same argument, in each round r the adversary can choose the graph G_r as a spanning tree where each ignorant agent is at a distance of at least 3 from the source, so by Lemma 17, broadcast is impossible. ◀

The above result does not hold for the trivial case of $d = 2$, since H_2 is simply a ring of four nodes where broadcast can be solved even for $k = 1$ (see Theorem 3). For a hypercube of dimension $d = 3$ (i.e., a cube) we can show a matching lower and upper bound of $k = n/2$ ignorant agents.

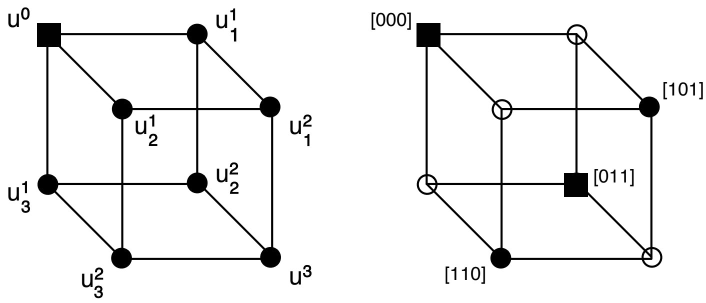

Figure 1 (a) The cube with a single source (denoted by a square) in the proof of Theorem 20. (b) The cube with two sources at distance 2; the remaining agents must occupy the two black nodes.

▶ **Theorem 20.** *If $G = H_3$ is a hypercube of dimension $d = 3$ consisting of $n = 2^3$ nodes, then $k = n/2$ ignorant agents are necessary and sufficient to solve broadcast.*

Proof (Lower Bound). We provide only a sketch of the proof that if there are only $k = 3$ ignorant agents, then it is not possible to solve broadcast starting from arbitrary configurations. In particular, we define a class QF of forbidden initial configurations, with one source and 3 ignorant agents in a cube, such that starting from any such configuration, the adversary can force the agents to move only to another configuration in QF. Further, in every configuration in QF, there is a spanning tree (defined by the available links) where the distance from source to the nearest agent is at least 3. The configurations in the set QF are listed below by showing the positions of the 3 agents, with respect to the source node which is always assumed[2] to be [000]:

[2] After any moves, we rename the nodes.

Q1 $([100], [010], [110])$
Q2 $([100], [010], [011])$
Q3 $([100], [110], [101])$
Q4 $([100], [101], [011])$
Q5 $([100], [110], [111])$
Q6 $([100], [011], [111])$
Q7 $([101], [011], [111])$
Note that starting from any other initial configurations with 3 agents permits a solution to broadcast. However, when the 3 ignorant agents start in any configuration isomorphic to configurations in QF, then the configuration in the next round can only be another configuration in QF. This implies that no ignorant agent can meet the source after any number of rounds, and thus it is not possible to solve the Broadcast problem. ◀

Proof (Upper Bound). We show that $k = 4$ agents can solve broadcast by providing an algorithm. We first show that one of the four agents can meet the source agent. We denote the nodes of the cube as follows: $u^0 = [000]$ is the node containing the source, u^1_1, u^1_2, u^1_3 are the three nodes at distance one from source, u^1_i means a generic node at distance 1 from the source, u^2_1, u^2_2, u^2_3 are the three nodes at distance 2 from the source, and $u^3 = [111]$ is the only node at distance 3 (see Figure 1(a)). We now consider all possible initial configurations with agents placed on distinct nodes; Note that, each such configuration has at least three empty nodes. We denote such a configuration as $[x, y, z, l]$ showing the positions of the 4 ignorant agents at nodes x, y, z, l (with $*$ denoting any node other than the source).

$C1$ Configuration $[u^1_1, u^1_2, u^1_3, *]$: At least one of the links (u^0, u^1_1), (u^0, u^1_2), or (u^0, u^1_3) must be active, otherwise node u^0 would be disconnected. Hence at least one agent can meet the source within the next time unit.

$C2$ Configuration $[u^2_1, u^2_2, u^2_3, *]$: At least one of the paths of distance two between the source node u^0 and one of the nodes u^2_1, u^2_2, or u^2_3 must be available, otherwise node u^0 would be disconnected from nodes u^2_1, u^2_2 and u^2_3. Hence, within the next step the agent at a distance two from the source, and the source agent move to the middle node of the path and meet.

$C3$ Configuration $[u^1_1, u^1_2, u^3, *]$: If at least one of the links (u^0, u^1_1) or (u^0, u^1_2) are active, then at least one agent can meet the source within the next time unit. Otherwise, the link (u^0, u^1_3) must be active (to ensure connectedness), so in that case, the source agent moves to node u^1_3. The configuration we obtain is isomorphic to the configuration $[c_2]$ above, so we are done.

$C4$ Configuration $[u^1_1, u^1_2, u^2_1, u^2_2]$: Assume that there are no paths of length 1 or 2 from source to any agent (otherwise we are done as explained above). In that case, any possible spanning tree must have the edges (u^0, u^1_3) and (u^1_3, u^2_3) and further at least one of the edges (u^1_2, u^2_3) or (u^2_1, u^3) or (u^2_2, u^3). In the first case, one agent moves to u^2_3 and we obtain the configuration $[c_2]$. In the other two cases, one agent moves to node u^3, and thus we obtain the configuration $[C3]$. So, we are done in all cases.

$C5$ Configuration $[u^1_i, u^2_1, u^2_2, u^3]$: In G_0, if there are no length-2 paths from source to any agent, then any path from u^0 must go through u^2_3; If node u^2_3 has an available edge to some agent, this agent will move to u^2_3 and we would obtain the configuration $[c_2]$. Otherwise u^2_3 has an available edge to some empty node $u^1_j \neq u^1_i$ which is connected to some node occupied by an agent. Thus, this agent moves to u^1_j, and we obtain the configuration $[C3]$.

We have shown one agent can reach the source within a constant number of steps and obtain the message. Now, there are two source agents and three ignorant agents. The two source agents can place themselves at distance two in G (this is always possible in at most 2 steps). Assume without loss of generality, that the two sources are at nodes [000] and [011] as in Figure 1(b). There are exactly two nodes ([110] and [101]) that are at distance 2 from both the sources. If the ignorant agents occupy these two nodes, then in any spanning tree chosen by the adversary, at least one of the ignorant agents would be at distance two from a source agent. And in fact, it is easy to see that either the three ignorant agents occupy all three adjacent nodes of one of the two sources (which means that in the next step at least one more agent will meet a source), or three ignorant agents occupy two adjacent nodes of each source. In the last case (due to connectedness) at least two of the ignorant agents will occupy the two nodes ([110] and [101]) in the next step. Finally, if only one of the ignorant agents occupies one of the nodes ([110] and [101]), then in the next step this agent can move from that node and therefore we obtain the previous configuration (i.e., where three ignorant agents occupy two adjacent nodes of each source). Thus eventually at least two of the ignorant agents will occupy the two nodes ([110] and [101]) and in the next round, at least one more ignorant agent will meet a source and therefore we will have 3 source agents.

Note that the case of 3 sources and 2 ignorant agents is analogous to the case of 2 sources and 3 ignorant agents, while the case of one ignorant agent and four sources, is analogous to the initial situation with one source and four ignorant agents. So, using the same strategies as above eventually all agents will obtain the message and broadcasting is solved. ◀

For hypercubes of higher dimensions $d \geq 4$, we do not have any general strategy for solving the problem as the adversary has too many possible ways of choosing the available subgraph. However, the lower bound of $k = n/2 - 1$ agents from Theorem 19 still holds and we have the upper bound of $k = n - 2$ (from Observation 1).

6 Conclusion

In this paper, we studied the problem of broadcast for mobile agents moving in constantly connected dynamic networks. The main objective is to understand how many agents are necessary and sufficient to allow broadcast to be solved in various topologies. It turns out that for sparse topologies such as rings and cactus graphs, the number of agents needed for solving the broadcast problem can be independent of the network size n, while for denser graphs including grids, hypercubes, as well as the complete graph, $\Theta(n)$ agents are needed. This preliminary investigation on broadcast in dynamic graphs opens many new research directions. For both grids and hypercubes, we have large gaps between the lower bounds of $(n - 2\sqrt{n})$ and $(n/2 - 1)$ respectively, and the upper bound of $(n - 2)$. It seems that solving the problem in grids requires more agents than in hypercubes, since grid networks contain more redundant edges. However, the lower bound on hypercubes shows that the number of agents needed can sometimes be much more than the number of redundant edges in the network. This is in contrast to the *cops and robbers* problem where the number of cops needed is roughly equal to the number of redundant edges in the underlying graph [3]. In the future, we would like to study the differences between various problems in this model and try to adapt techniques used for broadcast, to solve other problems in dynamic networks. Moreover it would be nice to classify various problems according to the resources needed for solving them under the adversarial model studied in this paper. Another possible direction of research would be to replace the strong assumption of global visibility with some weaker assumptions about the agent's capabilities that still suffices to solve broadcast in this model.

References

1 J. Anaya, J. Chalopin, J. Czyzowicz, A. Labourel, A. Pelc, and Y. Vaxès. Convergecast and broadcast by power-aware mobile agents. *Algorithmica*, 74(1):117–155, 2016.

2 B. Awerbuch and S. Even. Efficient and reliable broadcast is achievable in an eventually connected network. In *Proceedings of the 3th Symposium on Principles of Distributed Computing (PODC)*, pages 278–281, 1984.

3 S. Balev, J. J. Laredo, I. Lamprou, Y. Pigné, and E. Sanlaville. Cops and robbers on dynamic graphs: Offline and online case. In *Proc. Structural Information and Communication Complexity - 27th International Colloquium, SIROCCO 2020*, volume 12156 of *LNCS*, pages 203–219. Springer, 2020.

4 A. Casteigts, P. Flocchini, B. Mans, and N. Santoro. Shortest, fastest, and foremost broadcast in dynamic networks. *International Journal of Foundations of Computer Science*, 25(4):499–522, 2015.

5 A. Casteigts, P. Flocchini, W. Quattrociocchi, and N. Santoro. Time-varying graphs and dynamic networks. *International Journal of Parallel, Emergent and Distributed Systems*, 27(5):387–408, 2012.

6 M. Chrobak, L. Gasieniec, and W. Rytter. Fast broadcasting and gossiping in radio networks. *J. Algorithms*, 43(2):177–189, 2002.

7 A. Clementi, A. Monti, F. Pasquale, and R. Silvestri. Information spreading in stationary markovian evolving graphs. *IEEE Transactions on Parallel and Distributed Systems*, 22(9):1425–1432, 2011.

8 J. Czyzowicz, K. Diks, J. Moussi, and W. Rytter. Broadcast with energy-exchanging mobile agents distributed on a tree. In *Structural Information and Communication Complexity - 25th International Colloquium (SIROCCO)*, pages 209–225, 2018.

9 J. Czyzowicz, K. Diks, J. Moussi, and W. Rytter. Energy-optimal broadcast and exploration in a tree using mobile agents. *Theoretical Computer Science*, 795:362–374, 2019.

10 J. Czyzowicz, L. Gasieniec, A. Kosowski, E. Kranakis, D. Krizanc, and N. Taleb. When patrolmen become corrupted: Monitoring a graph using faulty mobile robots. *Algorithmica*, 79(3):925–940, 2017.

11 S. Das, R. Focardi, F. L. Luccio, E. Markou, and M. Squarcina. Gathering of robots in a ring with mobile faults. *Theoretical Computer Science*, 764:42–60, 2019.

12 S. Das, N. Giachoudis, F. L. Luccio, and E. Markou. Gathering of robots in a grid with mobile faults. In *45th International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM 2019)*, volume 11376 of *LNCS*, pages 164–178. Springer, 2019.

13 S. Das, G. A. Di Luna, and L. A. Gasieniec. Patrolling on dynamic ring networks. In *Proc. 45th Int. Conf. on Current Trends in Theory and Practice of Computer Science (SOFSEM)*, volume 11376 of *LNCS*, pages 150–163. Springer, 2019.

14 G.A. Di Luna, S. Dobrev, P. Flocchini, and N. Santoro. Live exploration of dynamic rings. In *Proocedings of the 36th IEEE International Conference on Distributed Computing Systems (ICDCS)*, pages 570–579, 2016.

15 G.A. Di Luna, P. Flocchini, L. Pagli, G. Prencipe, N. Santoro, and G. Viglietta. Gathering in dynamic rings. In *Proocedings of the 24th International Colloquium Structural Information and Communication Complexity (SIROCCO)*, pages 339–355, 2017.

16 T. Erlebach, M. Hoffmann, and F. Kammer. On temporal graph exploration. In *Proceedings of 42nd International Colloquium on Automata, Languages, and Programming (ICALP)*, pages 444–455, 2015.

17 A. Ferreira. Building a reference combinatorial model for manets. *IEEE Network*, 18(5):24–29, 2004.

18 P. Flocchini, B. Mans, and N. Santoro. On the exploration of time-varying networks. *Theoretical Computer Science*, 469:53–68, January 2013. doi:10.1016/j.tcs.2012.10.029.

19 L. Gasieniec. *Deterministic Broadcasting in Radio Networks*, pages 233–235. Springer US, Boston, MA, 2008. doi:10.1007/978-0-387-30162-4_105.

20 L. Gasieniec and A. Pelc. Adaptive broadcasting with faulty nodes. *Parallel Computing*, 22(6):903–912, 1996.

21 L. Gasieniec and A. Pelc. Broadcasting with linearly bounded transmission faults. *Discrete Applied Mathematics*, 83(1-3):121–133, 1998.

22 T. Gotoh, Y. Sudo, F. Ooshita, H. Kakugawa, and T. Masuzawa. Group exploration of dynamic tori. In *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*, pages 775–785, July 2018.

23 F. Harary and G. Gupta. Dynamic graph models. Mathematical and Computer Modelling, 25(7):79–88, 1997.

24 D. Ilcinkas, R. Klasing, and A.M. Wade. Exploration of constantly connected dynamic graphs based on cactuses. In *Proceedings 21st International Colloquium Structural Information and Communication Complexity (SIROCCO)*, pages 250–262, 2014.

25 D. Ilcinkas and A.M. Wade. On the power of waiting when exploring public transportation systems. In *Proceedings of the 15th International Conference on Principles of Distributed Systems (OPODIS)*, pages 451–464, 2011.

26 D. Ilcinkas and A.M. Wade. Exploration of the t-interval-connected dynamic graphs: the case of the ring. *Theory of Computing Systems*, 62(5):1144–1160, 2018.

27 F. Kuhn, N. Lynch, and R. Oshman. Distributed computation in dynamic networks. In *Proceedings of the 42nd Symposium on Theory of Computing (STOC)*, pages 513–522, 2010.

28 F. Kuhn and R. Oshman. Dynamic networks: Models and algorithms. SIGACT News, 42(1):82–96, 2011.

29 G. A. Di Luna. Mobile agents on dynamic graphs. In *Distributed Computing by Mobile Entities, Current Research in Moving and Computing*, pages 549–584. Springer, 2019.

30 O. Michail. An introduction to temporal graphs: An algorithmic perspective. *Internet Mathematics*, 12(4):239–280, 2016.

31 O. Michail and P.G. Spirakis. Traveling salesman problems in temporal graphs. *Theoretical Computer Science*, 634:1–23, 2016. doi:10.1007/978-3-662-44465-8_47.

32 R. O'Dell and R. Wattenhofer. Information dissemination in highly dynamic graphs. In *Proceedings of the Joint Workshop on Foundations of Mobile Computing (DIALM-POMC)*, pages 104–110, 2005.

33 D. Peleg and A. A. Schäffer. Time bounds on fault-tolerant broadcasting. *Networks*, 19(7):803–822, 1989. doi:10.1002/net.3230190706.

34 Y. Yamauchi, T. Izumi, and S. Kamei. Mobile agent rendezvous on a probabilistic edge evolving ring. In *2012 Third International Conference on Networking and Computing*, pages 103–112, December 2012.