Theoretical Computer Science 412 (2011) 1066–1080

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

The impact of mobility on the geocasting problem in mobile ad-hoc networks: Solvability and cost[☆]

R. Baldoni [a], A. Fernández Anta [b], K. Ioannidou [c], A. Milani [d,*]

[a] Dipartimento di Informatica e Sistemistica "Antonio Ruberti", Sapienza - Universitá di Roma, Roma, Italy
[b] Institute IMDEA Networks, Madrid, Spain
[c] Computer Science Department, University of California, Santa Cruz, CA, United States
[d] Labri, Université de Bordeaux 1, ENSEIRB, Talence, France

A R T I C L E   I N F O

Article history:
Received 19 April 2010
Received in revised form 6 October 2010
Accepted 3 December 2010
Communicated by R. Klasing

Keywords:
Mobile ad-hoc networks
Mobile distributed computing
Geocast
Geocast solvability vs. speed of movement
Speed of movement vs. cost of the solution
Distributed systems

A B S T R A C T

We present a model of a mobile ad-hoc network in which nodes can move arbitrarily on the plane with some bounded speed. We show that without any assumption on some topological stability, it is impossible to solve the geocast problem *deterministically* despite connectivity and no matter how slowly the nodes move. Moreover, even if each node maintains a stable connection with each of its neighbors for some period of time, it is impossible to solve the geocast problem if nodes move too fast. Additionally, we give a tradeoff lower bound which shows that the faster the nodes can move on a monodimensional space, the more costly it would be to solve the geocast problem. We provide geocasting algorithms for the case where nodes move in one dimension and also when they can move on the plane (i.e., in two dimensions). We prove correctness of our algorithms by giving exact bounds on the speed of movement. Our analysis helps understand the impact of speed of nodes, firstly, on geocasting solvability and, secondly, on the cost of geocasting.

1. Introduction

There has been increasing interest in mobile ad-hoc networks with nodes that move arbitrarily on the plane. This is justified by the significance of (wireless) mobile computing in emerging technologies. Current technologies require a stable infrastructure which is used for communication between mobile nodes. Unfortunately, in some cases, such as a military operation or after some physical disaster, a fixed infrastructure cannot exist. For such cases, it is desirable to program the mobile nodes to solve important distributed problems within specific geographical areas and without depending on a stable infrastructure. This is why there has been an increasing interest in studying problems related to the geographical position of participants in mobile ad-hoc networks such as georouting [3,4], geocasting [5–8], geoquorums [8], etc.

A fundamental communication primitive in certain mobile ad-hoc networks is *geocasting* [4]. This is an operation initiated by a node in the system, called the *source*, that disseminates some information to all the nodes in a given geographic area, named the *geocast region*. In this sense, the geocast primitive is a variant of multicasting, where nodes are eligible to receive[1]

☆ A preliminary version of this paper has appeared in DISC07 and OPODIS08. This research was supported in part by Comunidad de Madrid grant S2009TIC-1692 and Spanish MICINN grant TIN2008–06735-C02-01.

* Corresponding author. Tel.: +33 540003503.
E-mail addresses: baldoni@dis.uniroma1.it (R. Baldoni), antonio.fernandez@imdea.org (A. Fernández Anta), ioannidu@gmail.com (K. Ioannidou), milani@labri.fr (A. Milani).

1 Throughout this paper, we use the term deliver to refer to receiving the information being geocast. The term receive is used to describe the operation of receiving messages being broadcast while geocasting is taking place.

0304-3975/$ – see front matter
doi:10.1016/j.tcs.2010.12.006

the information according to their geographic location. While geocasting in two dimensions is clearly useful, geocasting in one dimension is also a natural operation in some real situations, like announcing an accident to nearby vehicles on a highway. In mobile ad-hoc environments, geocasting is also a basic building block to provide more complex services. As an example, Dolev et al. [8] use a deterministic reliable geocast service to implement atomic memory in mobile ad-hoc networks. A geocast service is *deterministic* if it ensures deterministic reliable delivery, i.e. all the nodes eligible to deliver the information will surely deliver it.

Designing a geocast primitive in a mobile ad-hoc network forces us to deal with uncertainty due to the dynamics of the network. Since communication links appear and disappear as nodes move in and out of the transmission range of other nodes, there is a (potential) continuous change of the communication topology. In other words, the movement of nodes and their speed of movement usually impacts on the lifetime of radio links. Then, since it takes at least $\Omega(\log n)$ time to ensure a one-hop successful transmission in a network with n nodes [9], mobility may be an obstacle for deterministic reliable communication.

This is because it can heavily influence, for example, the completion time of the message diffusion in a certain geographical area till making geocasting unsolvable if these speeds are high enough. In an extreme (unrealistic) scenario, nodes can move fast enough to ensure that no two neighbors stay connected for enough time to complete the receipt of a message. *Geocasting cannot be solved in this scenario even though the topology of the mobile ad-hoc network never disconnects*. To our knowledge this relation among problem solvability, the cost of a solution, and mobility has never been investigated. This paper focuses precisely on these issues. In particular, first, we provide a model of computation (Section 2) and a specification for the geocasting problem (Section 3) which both take into account (explicitly or implicitly) node mobility.

- The model does not rely on either GPS or synchrony. Instead, it relies on connectivity guarantees and restrictions on the movement speed and the geocast area to be able to solve the problem. The model makes a distinction between traditional connectivity and a new notion of connectivity, called *strong connectivity*. A strongly connected system has some assurance of topological stability, i.e., there is always a path between every two nodes formed by strong neighbors, where two nodes are strong neighbors if they remain neighbors for at least some period of time.
- The geocasting specification is split into three properties: reliable delivery, integrity, and termination. Reliable delivery states that all nodes, which remain for some positive time C within distance d from the location l where the geocast has been issued, will deliver the geocast information. Conversely, integrity defines the minimum distance between the location l and a node in order that the latter does not deliver the geocast information. Termination states that after some period of time C' from geocasting of some information, there will be no more communication related to this geocast.

In Section 4, the paper formally proves the following results related to solvability of geocasting:

- Even if the nodes are connected (in the traditional sense) and they move arbitrarily slow, it is not guaranteed that geocasting can be solved (Theorem 4);
- even if strong connectivity holds, then geocasting is still impossible for some bound on nodes' speed of movement (Theorems 5–7). Constraining the speed of movement below the bounds established in Theorem 5 is a necessary condition to ensure that one-hop communication succeeds. But this is not sufficient to ensure that the information propagates in space, i.e., the information may be transmitted from one node to the next, but never reaches a node in a given geographical position because the nodes covered by the information move in the opposite direction. This is proved in Theorem 6 Section 5.

Moreover, we provide lower bounds on the cost of the solution and we relate them with the mobility of nodes. In particular, in Theorem 8 we show a tradeoff lower bound that relates the completion time of geocasting to the speed of movement of nodes. We prove that the time complexity grows linearly with the speed of the movement of nodes and with the distance to be covered by the information. In a two dimensional space, the fact that the information could flow in one direction while it is needed in the other direction of the plane may just depend on adversarial communication topology. Observe that we do not assume knowledge concerning the network topology, except that it is strongly connected. Therefore, to ensure that the geocast is solved on a plane, the proposed algorithm has to cover all the nodes in the system. This is formalized in the lower bound presented in Theorem 9.

As a last contribution, if the speed is small enough, we show how to solve the geocasting problem in a one-dimensional setting (Section 6.1) and in a two-dimensional setting (Section 6.2). We prove that in a one-dimensional space the time complexity of these algorithms increases with the speed of nodes. In a two-dimensional space, following the previous result, the time complexity is dominated by the size of the system. None of the algorithms require any knowledge of the topology of the system and of the position of the nodes.

Finally, we would like to stress the fact that, to our knowledge, this is the first time that, in a mobile context, the computational complexity of a solution of a well-defined paradigm has been formally evaluated in terms of speed of nodes. Therefore, results in this paper can be considered as an important step towards understanding the "uncertainty" introduced by the mobility of nodes when solving distributed computing problems in a mobile setting. The steps we followed to analyze the impact of speed of nodes on problem solvability and on the cost of a solution can be followed by other researchers to formally prove correctness of their solutions of other distributed computing related problems in a mobile setting.

2. A mobile ad-hoc network model

We consider an unbounded set Π of (mobile) nodes which move in a continuous manner on a plane (2-dimensional Euclidean space) with bounded maximum speed v_{max}. The nodes in Π do not have access to a global clock, but their local clocks run at the same rate. Additionally, no node in Π fails.

Nodes communicate by exchanging messages over a wireless radio network. All the nodes have the same transmission radius r. At any time, each node is a neighbor of, and can communicate with, all the nodes that are within its transmission range at that time, i.e., the nodes that are *completely inside* a disk, centered at the node's position, of radius r [2]. Formally, let $distance(p, q, t)$ denote the distance between p and q at time t, we say that p and q are *neighbors* at time t if $distance(p, q, t) < r$. Nodes do not know their position, speed, nor direction of movement.

Local broadcast primitive. To directly communicate with their neighbors, nodes are provided with a *local reliable broadcast* primitive. Communication is not instantaneous, it takes some time for a broadcast message to be received. To simplify the presentation we consider as a time unit the *communication round* (or *round* for short), the time the communication of a message takes when accessing the underlying wireless network communication channel without collision. Additionally, we assume that local computation at the nodes takes negligible time (zero time for the purpose of the analysis). Since collisions are an intrinsic characteristic of MANETs, they have to be considered. We assume that the potential collisions due to concurrent broadcasts by neighbors are dealt by a lower level communication layer, and that this layer takes T rounds to (reliably and deterministically) deliver a message to its destination. The value of T could be related to the size of the system and depends on the complexity of the lower level communication protocol. As already stated, [9] shows that it takes at least $\Omega(\log n)$ time to ensure a one-hop successful transmission in a network with n nodes. Note that T is in $O(n)$ since a simple round robin algorithm would avoid any collisions in any possible network.

If a node p invokes $broadcast(m)$ at time t, then all nodes that remain neighbors of p throughout $[t, t + T)$ receive m by time $t + T$, for some fixed *known* integer $T > 0$. A node that receives a message m generates a $receive(m)$ event. It is possible that some node that has been a neighbor of p at some time in $[t, t + T)$ (but not during the whole period) also receives m, but there is no guarantee. However, no node receives m after time $t + T - 1$. A node issues a new invocation of the broadcast primitive only after it has completed the previous one (T time later). Then in each time interval of length T a node broadcasts at most one message.

Connectivity

The traditional definition of connectivity ensures that for every pair of nodes p and p' and every time t, there is at least one path of neighbors connecting p and p' at time t. Note that the traditional notion of connectivity states nothing about the expected stability of the communication topology. In fact, while maintaining the system connected, an adversary could continually change the neighborhood of nodes and make impossible even the basic task of geocasting (see Theorem 4 in Section 4).

For this reason, we introduce a stronger notion of connectivity, called *strong connectivity* which is based on the notion of *strong neighbors*. The latter is a dynamic concept aiming to capture the fact that if there is an upper bound on the speed of nodes, then the closer two neighbors are located to each other, the longer they will remain neighbors. Hence, if nodes are located fairly close, then their connection is guaranteed for some period of time.

Formally,

Definition 1 (*Strong Neighbor*)**.** Let $\delta_2 = r$ and δ_1 be fixed positive real numbers such that $\delta_1 < \delta_2$. Two nodes p and p' are *strong neighbors* at some time t if there is a time $t' \leq t$ such that $distance(p, p', t') \leq \delta_1$, and $distance(p, p', t'') < \delta_2$ for all $t'' \in [t', t]$.

Strong connectivity is a local property which helps to formalize the local stability in the communication topology necessary to solve the problem.[2]

Definition 2 (*Strong Connectivity*)**.** For every pair of nodes p and p' and every time t, there is at least one path of strong neighbors connecting p and p' at t.

When convenient, we may use the term that a pair of (strong) neighbors have a (strong) connection, or are (strongly) connected.

Two nodes p and p' *become* strong neighbors at time t if there is a $t' < t$, such that for any $t'' \in [t', t]$, p and p' are not strong neighbors at time t'', and p and p' are strong neighbors at time t.

Observe that once two nodes p and p' become strong neighbors (i.e., they are at distance δ_1 from each other), to get disconnected they must move away from each other so that their distance is at least δ_2. This means that the total distance to be covered in order for p and p' to disconnect is $\delta_2 - \delta_1$. We use the notation $\delta = \frac{\delta_2 - \delta_1}{2}$, where δ denotes the minimum

2 A different approach could be to constrain the mobility pattern of nodes (e.g. [11]) or to assume the global communication topology to be stable long enough to ensure reliable delivery (e.g. [16]).

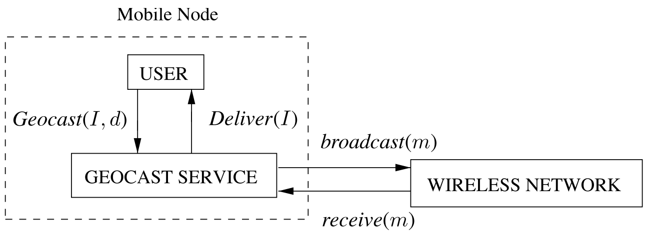


Fig. 1. System architecture.

distance that any two nodes that just became strong neighbors have to travel to stop being neighbors when moving in opposite directions. Thus, for a clear presentation of our results, we express the maximum speed of nodes, denoted v_{max}, as the ratio between δ and the time necessary to travel this space, denoted T'.

Definition 3 (*Movement Speed*)**.** Since $v_{max} = \frac{\delta_2-\delta_1}{2T'}$, it takes at least $T' > 0$ time for a node to travel distance $\delta = \frac{\delta_2-\delta_1}{2}$.

Since nodes move, the topology of the network may continuously change. In this sense, assuming both strong connectivity and an upper bound on the maximum speed of nodes provides some topological stability in the network. In particular, strong connectivity ensures that at least one neighbor of a node remains stable for some time. The period of stability depends on the maximum speed of nodes and on the parameters δ_1 and δ_2.

From the movement speed assumption, we derive that it takes at least $T' = \frac{\delta}{v_{max}} > 0$ rounds for a node to travel distance $\delta = \frac{\delta_2-\delta_1}{2}$ on the plane. Thus, from Definition 1 and the assumption on the bound on the speed, we gain some topological stability in the network, which is formally expressed in the following lemma:

Lemma 1. *If two nodes become strong neighbors at time t, then they are neighbors throughout the interval* $(t - T', t + T')$ *and remain strong neighbors throughout the interval* $[t, t + T')$.

Proof. If p and p' become strong neighbors at time t, then $distance(p, p', t) = \delta_1$. To be disconnected, they must move away from each other a distance of at least 2δ, so that their distance is at least δ_2. From Assumption 3, this takes at least T' time. Hence, for $\tau \in (t - T', t + T')$, $distance(p, q, \tau) < \delta_1 + 2\delta = \delta_2$, which proves the claims. □

3. The geocast problem

The geocast is a variant of the conventional multicasting problem, where nodes are eligible to deliver the information if they are located within a specified geographic region. The geocast region we consider is the circular area centered in the location where the source starts geocasting and whose radius is some given value d. We assume d to be provided as input by the user of the geocast primitive.

The geocast problem can be solved by a geocast service, implemented by a geocast algorithm which runs on mobile nodes. The geocast service provides two primitives to each mobile node: Geocast(I, d) to geocast information I at distance d and Deliver(I) to deliver information I. As illustrated in Fig. 1, on each mobile node there is a process running the geocast algorithm and a co-located user of the service which invokes geocast. The geocast algorithm uses *broadcast*(m) and *receive*(m) to achieve communication among neighbors.

3.1. A geocast specification

The geocast information is initially known by exactly one node, *the source*. If the source performs Geocast(I, d) at time t from location l, then:

Property 1 (*Reliable Delivery*)**.** *There is a positive integer C such that, by time* $t + C$*, information I is delivered (with Deliver(I)) to all nodes that are located at most at distance d away from l throughout* $[t, t + C]$.

The following properties rule out solutions which waste resources causing continuous communication or distribution of the information I to the whole Euclidean space.

Property 2 (*Termination*)**.** *If no other node issues another call to the geocast service, then there is a positive integer* C' *such that after time* $t + C'$*, no node performs any communication (i.e. a local broadcast) triggered by a geocast.*

Property 3 (*Integrity*)**.** *There is a* $d' \geq d$ *such that, if a node has never been within distance* d' *from l, it never delivers I.*

The parameters C, C', and d' of the above properties may depend on d and on the parameters of the system (like the number of nodes n, T, T', δ_2, or δ_1).

Observe that these properties are deterministic. This justifies the use of a deterministic reliable local broadcast primitive and the fact that we enforce nodes to be in range less than δ_2 during T steps to complete a successful communication.

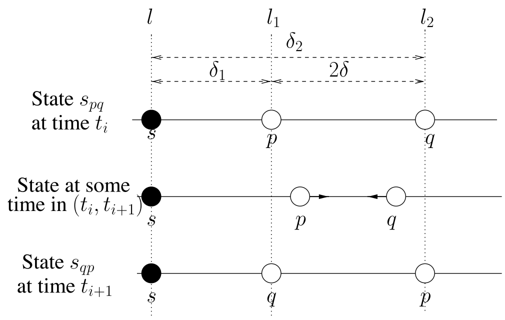


Fig. 2. Scenario for the proof of Theorem 5.

4. Problem solvability vs. node mobility

In this section, first we prove that traditional connectivity is not sufficient to solve the geocast problem, no matter how slowly the nodes move (Theorem 4). Then, we show several lower bounds that relate the speed of nodes' movement to solvability of a geocast. In particular, we prove that if the nodes move fast enough, then any geocast algorithm would fail even though strong connectivity is assumed. To prove our results, we describe executions that violate the reliable delivery property described in Section 3.

The following theorem shows an inherent limitation of traditional connectivity to model dynamic systems, where the network topology varies over time. In particular, it fails to guarantee some link stability which is on the other hand necessary to solve communication in mobile ad-hoc networks. Formally,

Theorem 4. *Even if the network is connected (but not strongly connected) and the nodes move arbitrarily slow, no Geocast(I, d) algorithm guarantees that the geocasting problem will be solved.*

Proof. Assume that the maximum speed of the nodes is $v > 0$. Consider a state, s_{pq}, such that all nodes are located on a straight line. The source s is the leftmost node at position l. The only neighbor, p, of s is on its right at distance $r - d_\epsilon$ from l, at position l_1 such that $d_\epsilon \le \frac{vT}{2}$. There is a node q located on the right of p at distance d_ϵ from p at position l_2.

Because $d_\epsilon \le \frac{vT}{2}$, distance $2d_\epsilon$ can be traversed by each node during T rounds. From state s_{pq} at time t, node q and p move with their maximum speed towards each other until they reach respectively location l_1 and location l_2 at time $t + \frac{T}{2}$. The state, s_{qp}, reached is the same as s_{pq} if we replace p by q and q by p, hence, connectivity is preserved. Then they immediately move towards each other with their maximum speed to reach the previous positions.

Because the switch between s_{pq} and s_{qp} and then again s_{pq} takes T rounds, no matter when the source node will broadcast the message with information I, neither p nor q will remain its neighbor long enough to reliably receive the message. We can repeat the scenario infinitely many times. □

Lower bounds. For the following results we assume strong connectivity and we show that if the speed of movement of nodes is too high, the geocast problem cannot be solved.

Recall that a link between two nodes is created or destroyed when a node enters or exits the transmission range of the other node, respectively. Moreover, for one-hop communication to succeed, a time of period of T time units during which a link is stable is required. Thus, if the nodes move fast enough then the link between any two nodes may not last enough time for a one-hop communication to succeed (Theorem 5). In particular, our lower bound relates the stability of a link, expressed in terms of the speed of movement of nodes, with the time necessary to a single-hop transmission to succeed.

Theorem 5. *No algorithm can solve the geocast problem in one dimension if $v_{max} \ge \frac{\delta_2 - \delta_1}{T}$, i.e. if $T' \le \frac{T}{2}$ even if strong connectivity holds.*

Proof. Consider a *Geocast*(I, d) primitive invoked at some time t by a source s, with $d \ge \delta_2$. We prove the claim by presenting a scenario in which, independently of the algorithm used, no node except the source delivers information I, while there are other nodes in the geocast region permanently. This violates the reliable delivery property and hence the geocast problem is not solved.

In our scenario there are three nodes, the source s and nodes p and q, that are permanently in the geocast region. Initially, node s is at a position l, from which it will never move. Node p is at position $l_1 = l + \delta_1$ (at a distance δ_1 from s), and q is at a distance δ_1 from p and at a distance $2\delta_1$ from s. Then, q moves to reach the state s_{pq} depicted in Fig. 2, which has the following properties: all nodes are located on a single line; the leftmost node is the source s located at position l; a node p is located at position l_1 at distance δ_1 from l; and a node q is located at position l_2 at distance 2δ from l_1. Observe that from the initial configuration up to state s_{pq} strong connectivity holds, and that nodes p and q are always within distance $\delta_2 \le d$ of l.

If s never broadcasts I then neither p nor q deliver it, and reliable delivery is violated. Otherwise, assume that as a consequence of the *Geocast*(I, d) invocation, s invokes *broadcast*(I) at times $t_0, t_1, \ldots$, with $t_{i+1} \ge t_i + T$. Let us define first the behavior of the nodes in interval $[t_0, t_1]$. At time t_0 nodes p and q start moving at their maximum speed v_{max} to

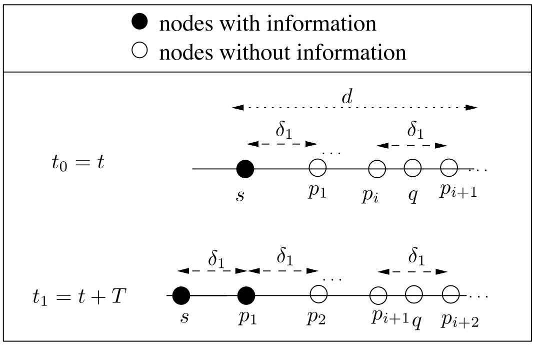


Fig. 3. Proof of Theorem 6.

exchange their positions. Then, at time $t_0' = t_0 + 2T'$, p is located at l_2 and q is located at l_1 reaching state s_{qp}. They do not move from that state until t_1. Observe that strong connectivity has been preserved during the whole period $[t_0, t_0']$: p and q never stop being strong neighbors, and the source is a strong neighbor of p for all the period $[t_0, t_0')$ and at time t_0' it becomes a strong neighbor of q. Note that neither p nor q have been neighbors of s during the whole period $[t_0, t_0 + T]$, because q is not a neighbor at time t_0 and p is not a neighbor at time $t_0' \leq t_0 + T$. Hence, in our execution no node delivers I in $[t_0, t_1]$.

The behavior in interval $[t_1, t_2]$ is the same as described but exchanging the roles of p and q: the initial state is s_{qp}, at time t_1 they start moving to exchange positions, and at time t_1' they end up at state s_{pq}. Again, I is not delivered at p nor q because they have not been neighbors of s in the whole period $[t_1, t_1 + T]$. For any interval $[t_i, t_{i+1}]$ the behavior is the same as in interval $[t_0, t_1]$, if i is even, and the same as in interval $[t_1, t_2]$ if i is odd. Then, in this scenario of execution only s delivers I and the reliable delivery property is not fulfilled. □

In Theorem 6, we prove that even if one-hop communication succeeds, (i.e., a node is able to transfer the information to at least one of its neighbors), mobility can cause limited propagation of the information. In particular, we show a scenario for which the geocast information does not propagate in space towards a given direction where there is a (fixed) node that eventually should receive it. This is achieved by moving all nodes that have delivered the information away from the fixed node. In particular, we provide a scenario in which each node that delivers the information, delivers it when being in the exact position l where the geocast(I, d) has been invoked, while there is a node on the right of this location that should deliver I. Thus, the information does not move towards the right of l and a node at within distance d from l does not deliver I, thus violating the reliable deliver property.

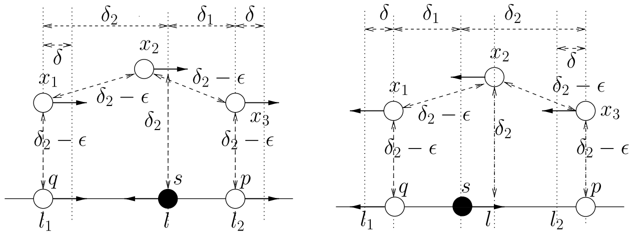
Theorem 6. *For $\delta_1 \geq \frac{\delta_2}{2}$, no Geocast($I$, d) algorithm can solve the geocast problem if $v_{max} > \frac{\delta_1}{T}$ for a system with an infinite number of nodes even if strong connectivity holds.*

Proof. We describe an execution (illustrated in Fig. 3) during which all nodes are placed on a straight line and a node receives a message containing I if and only if it is located on or on the left of the original location, l, of the source $s = p_0$. In this execution, there is a node, q, always located on the right of this position at a distance less than d, and hence, never delivers I, violating reliable delivery. Initially, at time $t = t_0$, the nodes are placed on a line on the right of q_0, one every δ_1 distance, with the exception of q. Let p_i be the node located at distance $i\delta_1$ on the right of l at time t_0 (for $i \geq 0$). At time t_0, the only neighbors of p_0 are p_1 and possibly q because, since $\delta_1 \geq \frac{\delta_2}{2}$, all other nodes are at distance at least δ_2 from p_0. Similarly, at time t_0, the only neighbors of p_i (for $i \geq 1$) are p_{i-1}, p_{i+1} and possibly q. All nodes p_i for ($i \geq 0$) move continually, with speed $\frac{\delta_1}{T}$ towards the left. Note that this is possible because by our assumption $v_{max} > \frac{\delta_1}{T}$ (i.e., $\frac{\delta_1}{T}$ is smaller than the maximum speed v_{max}). All other nodes p_i for $i \geq 0$ form a path such that each two consecutive nodes are strong neighbors. Furthermore, q is always a strong neighbor of the first node on its right throughout the execution because their distance is at most equal to δ_1. We conclude that strong connectivity holds.

At time t_0 only p_0 (at location l) knows I. Node p_1 delivers I at time $t_1 = t + T$ when it is at location l. This is because during T rounds, p_1 moves a distance $\frac{T\delta_1}{T} = \delta_1$ and it moves towards the left starting from a location at a distance δ_1 on the right of l. At time t_1, both p_0 and p_1 will rebroadcast messages with information I. Similarly, node p_i is the rightmost node to deliver I at time $t_i = t + iT$ when at location l. All other nodes that delivered I are on the left of location l at that time. Since q is never a neighbor of any node on or on the left of position l, it will never deliver I. □

Solving geocast in two dimensions. The previous bounds were proved using executions where nodes move on a line. Thus they still hold if nodes have more degree of freedom of movement, i.e., they can move in the plane. Additionally, we show that if nodes move on a plane, T' must be larger than T for the geocast problem to be solvable.

Theorem 7. *No algorithm can solve the geocast problem if nodes move in two dimensions if $v_{max} \geq \frac{\delta_2 - \delta_1}{2T}$, (i.e. if $T' \leq T$) even if strong connectivity holds.*



(a) State at time t and t_0.

(b) State at time $t_0' = t_0 + T^v$.

Fig. 4. Scenario for the proof of Theorem 7.

Proof. Consider a *Geocast*(I, d) primitive invoked at some time t by a source s, with $d \geq \delta_2$. To prove the claim we construct a scenario with 6 nodes, and an execution in it, such that the geocast region contains several nodes permanently, but only s delivers I. Since the reliable delivery property is not satisfied, this proves the claim.

In our scenario, there are 6 nodes, the source s, and nodes p, q, x_1, x_2, and x_3. At the time t of the geocast, we assume that the system is in the state shown in Fig. 4(a). This state can be reached from an initial situation in which the nodes q, x_1, x_2, x_3, p, and s are placed (in this order) on a line, at distance δ_1 of each one from the next, and move without breaking the strong connectivity, to the state of Fig. 4(a). Observe that all nodes are strongly connected along the path q, x_1, x_2, x_3, p, s, but that the source is not a neighbor of neither x_1, x_2, nor x_3. Additionally, both p and q are in the geocast region (since $d \geq \delta_2$). They will be in the region during the whole execution and hence to satisfy reliable delivery they should deliver I.

Let us first assume then that, although the invocation to *Geocast*(I, d), s never makes a call to *broadcast*(I). Then, p and q will never receive, and hence deliver, I and reliable delivery is violated. Otherwise, assume that as a consequence of the *Geocast*(I, d) invocation, s invokes *broadcast*(I) at times $t_0, t_1, \ldots$, with $t_{i+1} \geq t_i + T$. Let us define first the behavior of the nodes in interval $[t_0, t_1]$. At time t_0, the source s and node q start moving towards each other at the maximum speed v_{max}, while nodes p, x_1, x_2, and x_3 start moving in the same direction as q. At time $t_0' = t_0 + T' \leq t_1$ all nodes have traveled a distance of δ (by definition of T') and the system is in the state depicted in Fig. 4(b). In the interval $[t_0', t_1]$ no node moves.

Observe that strong connectivity has been preserved during the whole period $[t_0, t_0']$, since the distances along the path q, x_1, x_2, x_3, p did not change, and the source is a strong neighbor of p for all the period $[t_0, t_0')$ and at time t_0' it becomes a strong neighbor of q. Note also that neither p nor q have been neighbors of s during the whole period $[t_0, t_0 + T]$, because q is not a neighbor at time t_0 and p is not a neighbor at time $t_0' \leq t_0 + T$. Hence, in our execution no node delivers I in $[t_0, t_1]$.

The behavior in interval $[t_1, t_2]$ is the same as described for $[t_0, t_1]$, but swapping the directions of movement and the roles of p and q. The initial state at time t_1 is the one show in Fig. 4(b), and the final state reached at time $t_1' = t_1 + T'$ is the one shown in Fig. 4(a). Again, I is not delivered at p nor q because they have not been neighbors of s in the whole period $[t_1, t_1 + T]$. For any interval $[t_i, t_{i+1}]$ the behavior is the same as in the interval $[t_0, t_1]$, if i is even, and the same as in interval $[t_1, t_2]$ if i is odd. Then, in this scenario of execution only s delivers I and the reliable delivery property is not satisfied. □

5. The cost of geocasting vs. the speed of nodes

In this section, we show how the speed of movement of nodes relates to the cost of any Geocast algorithm. From Theorem 5, we know that the problem cannot be solved if $v_{max} \geq \frac{2\delta}{T}$ which is equivalent to $T' \leq \frac{T}{2}$. From Theorem 6, we know that the problem cannot be solved if $v_{max} > \frac{\delta_1}{T}$ which is equivalent to $T' < \frac{\delta T}{\delta_1}$. Hence to solve the problem it is necessary that $T' > \max\{\frac{1}{2}, \frac{\delta}{\delta_1}\}T$ but this may be not sufficient. Theorem 8 verifies the intuition that if a solution to a geocast exists, then the time complexity of this solution is proportional to the speed of movement of the nodes. In particular, the larger the speed of the nodes can be (which is inversely related to T') the more time it would take to solve geocasting.

Theorem 8. *Assuming that $T' > \frac{T}{2}$ and $\delta_1 \geq \frac{\delta_2}{2}$, then if it is possible to solve the geocast problem, it would take more than $(\lfloor \frac{d-\delta_2}{\delta_1 - \frac{T\delta}{T'}} \rfloor + 1)T$ rounds to ensure reliable delivery, using any Geocast(I, d) algorithm for a system with more than $\lfloor \frac{d-\delta_2}{\delta_1 - \frac{T\delta}{T'}} \rfloor$ nodes even if strong connectivity holds.*

Proof. We describe an execution (illustrated in Fig. 5) of a geocast algorithm that causes as much rebroadcasting as possible and which cannot guarantee reliable delivery in less than $(\lfloor \frac{d-\delta_2}{\delta_1 - \frac{T\delta}{T'}} \rfloor + 1)T$ rounds. During this execution there is a node, q, located exactly at distance d from the original location, l, of the source, $s = p_0$. At time t_0, the nodes (other than q) are placed on a line on the right of p_0, one every δ_1 distance. Let p_i be the node at distance $i\delta_1$ on the right of l at time t_0 (for $i \geq 0$). At time t_0, the only neighbors of p_0 are p_1 and possibly q because (since $\delta_1 \geq \frac{\delta_2}{2}$) all other nodes are at distance at least δ_2 from p_0. Similarly, at time t_0, the only neighbors of p_i (for $i \geq 1$) are p_{i-1}, p_{i+1} and possibly q. All nodes p_i for ($i \geq 0$) move continually, with their maximum speed (i.e., $\frac{\delta}{T'}$) towards the left. Strong connectivity holds because, all nodes p_i (for

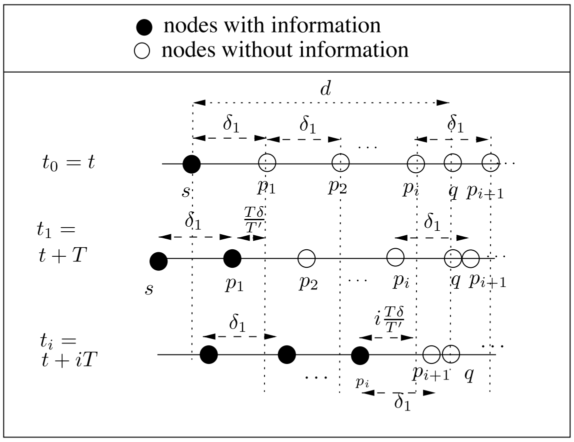


Fig. 5. Proof of Theorem 8.

$i \geq 0$, other than q) form a path of strong neighbors and q is a strong neighbor of the first node on its right throughout the execution because their distance is at most equal to δ_1.

At time $t = t_0$ only p_0 knows I. Node p_1 first delivers I at time $t_1 = t + T$ when it is at a distance $\delta_1 - \frac{T\delta}{T'}$ on the right of l. Node p_i is the rightmost node to deliver I at time $t_i = t + iT$ when at a distance $i\delta_1 - i\frac{T\delta}{T'}$ on the right of l. Node q can only deliver I within T rounds after at least one of its neighbors has delivered I. The earliest this happens is within T rounds after I is delivered by a neighbor of q on its left. This neighbor has to be at distance smaller than δ_2 from q. Hence, reliable delivery cannot happen before time $t_j + T (= t + (j + 1)T)$ for the smallest possible j for which $d - (j\delta_1 - j\frac{T\delta}{T'}) < \delta_2$ (i.e., $j > \lfloor \frac{d-\delta_2}{\delta_1 - \frac{T\delta}{T'}} \rfloor$). □

Finally, assuming an upper bound, n, on the number of nodes in the system we show that there are cases in which any geocast algorithm requires $\Omega(nT)$ time to complete if nodes can move on a plane.

Theorem 9. *Any deterministic Geocast(I, d) algorithm (if it exists) that solves the geocast problem in two dimensions requires $\Omega(nT)$ time to complete even if strong connectivity holds.*

Proof. Consider an execution where at time t there is a node p within distance d but greater than δ_2 from the source node s. So at time t nodes p and s are not neighbors.

At time t all the other nodes are located in a chain shaped as a reversed U, such that this chain connects s to p and each pair of nodes is at distance $\delta_2 - \epsilon$ from its neighbors in the chain. Before we reach this configuration, any pair of neighbors was previously at distance δ_1 from each other. Thus, strong connectivity holds from time t onward. At the same time t, a call of geocast is invoked at the source node. The information moves one-hop farther in the chain any T times. Since the chain is composed by n nodes, the claim follows. □

The bound of the above theorem depends on n. If n is finite this bound is finite. However, in a system with potentially infinite nodes, the geocast problem may never be solved.

Corollary 1. *There is no deterministic Geocast(I, d) algorithm that can solve geocasting for a system with infinite nodes that move on a plane.*

Note that the above Corollary does not hold when nodes move on a one dimensional space. In fact, in the following section we provide an algorithm that solves geocast, even though the system has infinite nodes, provided that nodes do not move too fast.

6. A framework for geocasting algorithms

In this section we present a natural class of algorithms, denoted *M-Geocast*(I, d). Later we show that when M is instantiated with the appropriate value, the geocast problem can be solved.

Algorithm *M-Geocast*(I, d) works as follows. When the source node invokes a call *Geocast*(I, d), it immediately delivers the information I (Line 8). Then, it broadcasts a message $[I, 0]$ and stores in a local variable *TLB* the time this first transmission happened (Lines 10–11), in order to retransmit every T units of time (Lines 13–14). When a node p receives for the first time a message $[I, count_I]$, it immediately delivers I. Then, it starts broadcasting messages with information I periodically (Lines 2–6).

Observe that in Algorithm *M-Geocast*(I, d) nodes exchange pairs of values, formed by the information I and a counter $count_I$.[3] The value $count_I$ contains an estimate of the time that has passed since the geocast started. This value, combined

3 For simplicity we will often say that they receive and (re)broadcast information I.

with the the parameter M, is used to terminate the algorithm. The source node sets this counter to 0 initially. Since it takes at least one round for a node p to receive a message $[I, count_I]$ broadcast by a node q, by incrementing the counter $count_I$ before rebroadcasting, node p notifies the next receiver that one more round has elapsed from the initial invocation of the geocast. Note that the counter may not be accurate, since at least one but at most T rounds could have passed. Hence, although the use of this counter improves message complexity, it does not necessarily optimize it. In particular, if there was a way to know if more than one rounds have elapsed, then the counter could increase faster, causing the predicate to evaluate to false faster. This would result in less messages to be broadcast without affecting the correctness of the algorithm.

```
Init
    TLB_I ← ⊥

upon event ⟨receive([I, c])⟩
    if (TLB_I = ⊥) then
        trigger ⟨Deliver(I)⟩;
        count_I ← c + 1;
        trigger ⟨broadcast([I, count_I])⟩
        TLB_I ← clock
    end if

Procedure M-Geocast(I, d)
    trigger ⟨Deliver(I)⟩;
    count_I ← 0;
    trigger ⟨broadcast([I, count_I])⟩;
    TLB_I ← clock

when (clock = TLB_I + T) and (count_I < M)
    count_I ← count_I + T;
    trigger ⟨broadcast([I, count_I])⟩
    TLB_I ← TLB_I + T
```

Fig. 6. The code of the *M-Geocast*(I, d) algorithm class.

The algorithm presented in Fig. 6 represents a class of very natural solutions for geocasting. The difficulty lies in proving correctness if strong connectivity holds after specifying appropriate values of M. In the following subsections, we provide such solutions and their formal proofs by giving exact values to M that suffice for correctness of the algorithm separately for the case of nodes moving in one dimension or in two dimensions (i.e., on a plane). In the rest of this section we assume that strong connectivity holds, i.e., all pairs of nodes in the network are strongly connected.

6.1. Algorithm for the one-dimensional case

Assume that the source $s = q_0$ initiates a call of *M-Geocast*(I, d) at time $t = t_0$ from location $l = l_0$. Next, we prove that I propagates from l_0 towards the right of l_0. (For the left of l_0, the proof is symmetrical.) This happens in steps so that within a small period of time, I moves from a node, q_j to another node q_{j+1} at some large distance away.

Observation 1. *Let p be a node that receives information I at time t, then either p immediately rebroadcasts I or it exists at time $\tau \in [t, t+T]$ such that p broadcasts I both at time $\tau - T$ and at time τ.*

Observation 2. *If $T' > T$, $\frac{\delta T}{T'} < \delta$ is the maximum distance that a node can cover in time T.*

Observation 3. *Let p be a node that receives a message with information I at time t. p has delivered information I by time t.*

Hereafter, we denote $\Delta = \delta_1 + \delta$.

Lemma 2. *Let q_j be a node located at location l_j at time t_j. If $T' > T$ then every node that at time $t_j + T$ is within distance $\Delta = \delta_1 + \delta$ from l_j has been a neighbor of q_j throughout all the period $[t_j, t_j + T]$.*

Proof. At time t_j, q_j is located in l_j and it is a neighbor of all nodes at a distance minor than δ_2 from l_j. Let p be a node that at time $t_j + T$ is located within a distance Δ from l_j. Let v be the maximum speed of the nodes, since $T' > T$, in T time a node can travel at most a distance $vT = \frac{\delta}{T'}T < \delta$. Thus at time t_j, p was located at l_p within distance $\Delta + vT < \delta_2$ from l_j.

To break the connection with p, q_j has to travel in the opposite direction of p at some time during $[t_j, t_j + T]$. Without loss of generality, consider q_j to move towards the left. At any time $t \in [t_j, t_j + T]$, q_j will be located at $l_j - vt$ and p will be at $l_p + vt$. Let l_p^T denote the location of p at time $t_j + T$, $l_p^T = l_p + vT$. Then, $l_p = l_p^T - vT$ and for all $t \in [t_j, t_j + T]$ $l_p^t = l_p + vt = l_p^T - vT + vt = l_p^T + v(t - T)$. So at time t the distance between q_j and p is $distance(p, q_j, t) = l_p^t - l_q^t = l_p^T + v(t - T) - (l_j - vt) = l_p^T - l_j + 2vt - vT \leq l_p^T - l_j + vt$. Since $l_p^T \leq l_j + \delta_1 + \delta$, $distance(p, q_j, t) \leq \delta_1 + \delta + vt < \delta_2$ because $vt < \delta$ for all $t \in [t_j, t_j + T]$. □

Lemma 3. *Let a node q broadcast information I at time t being at location l. If $T' > T$, then every node that at time $t + T$ is within distance Δ from l will deliver the information by time $t + T$.*

Proof. By Lemma 2, every node p that at time $t + T$ is within distance Δ from l is a neighbor of q throughout all the period $[t, t + T]$. Thus if q broadcasts the information I at time t, p will deliver I by time $t + T$. □

The following Lemma 4 states that if a node exists that broadcasts information I at some time t, then by time $t + 3T$ there is another node far away from location l which broadcasts information I. Thus these two nodes define a non-zero spatial interval and a temporal interval between two successive broadcasts events.

Lemma 4. *Let a node q_j broadcast the information I at time t_j located at point l_j. Let L_r denote the set of nodes that at time t_j are located on the right of $l_j + \delta_1$. If $\delta_1 \geq \delta$, $T' > T$ and $L_r \neq \emptyset$ then, assuming that $count_I < M$ for all nodes throughout $[t_j, t_{j+1}]$, either all the nodes in L_r deliver information I by time $t_j + T$ or there is a node q_{j+1} which broadcasts I at time t_{j+1} at location l_{j+1} such that:*

1. $t_{j+1} - t_j \leq 3T$,
2. $l_{j+1} - l_j \geq \delta_1 - \frac{\delta T}{T'} > 0$
3. *let $t = \min(t_j, t_{j+1} - T)$, throughout all the interval $[t, t_{j+1}]$, node q_{j+1} is a neighbor of another node q located on the left of $l_j + \Delta$ and which invoked broadcast$([I, \cdot])$ at some time in $[t_{j+1} - T, t_{j+1}]$.*

Proof. Assume that at time t_j, a node q_j broadcasts the information I being located at l_j. One of the following two cases holds:

- At time $t_j + T$, there is at least a node p located in the interval $[l_j + \delta_1, l_j + \Delta]$. Then by Lemma 3 p will deliver the information I by time $t_j + T$. By Observation 1, p will broadcast I at some time $t_{j+1} \in [t_j, t_j + T]$. By Observation 2 and its position at time $t_j + T$, p will rebroadcast I at time $t_{j+1} \leq t_j + T$ being at location $l_{j+1} \geq l_j + \delta_1 - \frac{\delta T}{T'}$. The claim holds being $q_{j+1} = p$ and $q = q_j$.
- At time $t_j + T$, no node is located in the interval $[l_j + \delta_1, l_j + \Delta]$. Let L and L' respectively denote the set of nodes that at time $t_j + T$ are located on the left of $l_j + \delta_1$ and the ones that at time $t_j + T$ are located on the right of $l_j + \Delta$.

  If $L' = \emptyset$ then all the nodes that at time t_j were on the right of $l_j + \delta_1$ are within distance Δ from l_j at time $t_j + T$. By Lemma 3, these nodes deliver information I by time $t_j + T$.

  Otherwise, there must exist paths of strong neighbors from nodes in L' to node on the left of $l_j + \delta_1$. In particular nodes $\in L'$ can be connected with nodes in L at most within distance δ on the left of l_j. These latter have delivered the information I by time $t_j + T$. One of the following cases has to hold:

  1. There exists at least a connection between a node p in L' and a node q in L which lasts throughout $[t_j, t_j + 2T]$. Then p will deliver the information I at some time $t \in [t_j, t_j + 2T]$. Note that at time $t_j + T$, p is on the right of $l_j + \Delta$ and since $T' > T$ it is on the right or on $l_j + \Delta - \frac{\delta T}{T'} > l_j + \delta_1 - \frac{\delta T}{T'}$ throughout all the period $[t_j + T, t_j + 2T]$. Then by Observation 1, p will broadcast information I at some time $t_{j+1} \in [t_j + T, t_j + 2T]$ being located at some position $l_{j+1} > l_j + \delta_1 - \frac{\delta T}{T'}$. The claim holds being $q_{j+1} = p$ and by the fact that p and q are neighbors throughout $[\tau, t_{j+1}] \subset [t_j, t_j + 2T]$, where $\tau = \min t_{j+1} - T, t_j$.
  2. Each connection between nodes in L' and nodes in L breaks at some time in $[t_j, t_j + 2T]$. Then, a new strong connection has to be created at some time $t \in [t_j, t_j + 2T]$ before all such connections break. Otherwise strong connectivity is violated, which is a contradiction.

     Let p and q be respectively the node in L and the node in L' that create the new strong connection at time t, i.e. $distance(p, q, t) \leq \delta_1$. By Lemma 2 p and q have been neighbors throughout $[t - T, t + T]$. If $t \in [t_j, t_j + T]$, $[t_j, t_j + T] \subset [t - T, t + T]$ and all such connections have to break at some point in $[t_j + T, t_j + 2T]$ otherwise there will exist at least a connection between a node in L and a node in L' that lasts throughout all the period $[t_j, t_j + 2T]$ and thus we reach a contradiction.

     Then a new connection between a node p in L and a node q in L' has to be created at some time $t \in [t_j + T, t_j + 2T]$. At time t, $distance(p, q, t) \leq \delta_1$, and since in $2T$ time a node can travel at most a distance $\frac{2\delta T}{T'}$, at time t_j, q was on the right of $l_j - 2\delta$. Thus q delivers information I by time $t_j + T$, and q broadcasts I both at time τ and $\tau' = \tau + T$ with $\tau \in [t_j, t_j + T]$. By Lemma 1, p and q are neighbors throughout all the period $[t - T, t + T]$ with $t \in [t_j + T, t_j + 2T]$. Either τ or τ' is in the interval $[t - T, T]$, then p delivers information I by time $t + T \leq t_j + 3T$. Then either p immediately broadcasts I or it broadcasts I at some time in $[t, t + T]$. Then p broadcasts information I at time $t_{j+1} \leq t_j + 3T$ being at some location $l_{j+1} > l_j + \delta_1 + \delta - \frac{2\delta T}{T'} > l_j + \delta_1 - \frac{\delta T}{T'}$. The claim holds being $q_{j+1} = p$ and by the fact that throughout $[t_{j+1} - T, t_{j+1}] \subset [t - T, t + T]$ p and q are neighbors. □

Observation 4. *Let q, q' and p be three nodes that at time t are respectively located at l_q, $l_{q'}$ and l_p such that $l_p < l_q$ and $l_p < l_{q'}$. Let q deliver information I by time $t + T$ because p invoked a call of broadcast$([I, \cdot])$ at time t. If q' is between q and p throughout $[t, t + T]$, q' delivers information I by time $t + T$.*

Definition 4. t_j is a time at which a node q_j invokes *broadcast*$([I, \cdot])$ being located at location l_j such that $t_{j+1} - t_j \leq 3T$ and $l_{j+1} - l_j \geq \delta_1 - \frac{\delta T}{T'}$, for $j \in \{0, 1, \ldots, i\}$.

Lemmata 5–7 are instrumental to prove Lemma 8. This latter states that any node that traverses any of the spatial intervals defined by two consecutive broadcast events (the ones defines in Definition 4) during the corresponding broadcast period, delivers the information by a given time.

Lemma 5. *Let $t, t' \in [t_j, t_{j+1}]$ with $t' > t$. Let p be a node that at time t is on the left of l_j and at time t' is located inside the interval $[l_j, l_{j+1}]$. If $\delta_1 > \delta$, p receives a message with information I by time $t_j + T$.*

Proof. By Definition 4, $t_{j+1} - t_j \leq 3T$. Let $t, t' \in [t_j, t_{j+1}]$ with $t' > t$. Let p be a node that at time t is on the left of l_j and at time t' is located inside the interval $[l_j, l_{j+1}]$. If $\delta_1 > \delta$, at time t_j p is at most within distance $3\delta < \delta_2$ on the left of l_j. Then, at time t_j, q_j and p are neighbors and they will remain neighbors at least up to $t_j + T$. This is because in the worst case p reaches position l_j immediately after t_j but then at time $t_j + T$ the distance between p and q_j is at most 2δ. Otherwise they move towards each other getting closer. So p will receive a message with information I by time $t_j + T$. □

Lemma 6. *Let $t, t' \in [t_{j+1} - T, t_{j+1}]$ with $t' > t$. Let p be a node that at time t is on the right of location l_{j+1}. If $\delta_1 > \delta$, p receives a message with information I by time $t_{j+1} + T$.*

Proof. Let $t, t' \in [t_{j+1} - T, t_{j+1}]$ with $t' > t$. Let p be a node that at time t is on the right of location l_{j+1}. If at some time $t' \in [t_{j+1} - T, t_{j+1}]$ p is on the left of l_{j+1}, p is neighbor of q_{j+1} throughout all the period $[t_{j+1}, t_{j+1} + T]$. This is because $\delta_1 > \delta$ and at time t_{j+1}, p is on the right of $l_{j+1} - \delta$ and in T times the distance between p and q_{j+1} increases less than 2δ. Then p will receive a message with information I by time $t_{j+1} + T$. □

Lemma 7. *Let p be a node that at time $t \in [t_j, t_{j+1}]$ is on the right of l_{j+1}. If at some time $t' \in [t_j, t_{j+1}]$ with $t < t'$, p is located at $l_p \in [l_j, l_{j+1}]$ and there does not exist a time $t'' \in [t_j, t_{j+1}]$ with $t'' > t'$ such that p is not on the right of l_{j+1}, p delivers the information I by time $t_{j+1} + 2T$.*

Proof. Consider a node p that at time $t \in [t_j, t_{j+1}]$ is located at the right of location l_{j+1}. Assume that at time $t' \in [t_j, t_{j+1}]$, with $t' > t$, p is located inside the interval $[l_j, l_{j+1}]$ and $t'' \in [t_j, t_{j+1}]$ does not exist with $t'' > t'$ such that p is on the right of l_{j+1} at t''.

If $t' \in [t_{j+1} - T, t_{j+1}]$, the claim follows by Lemma 6 and Observation 3. Then, consider $t' \in [t_j, t_{j+1} - T)$. $[t_j, t_{j+1} - T) \subseteq [t_j, t_j + 2T)$, then by Observation 2, at time $t_{j+1} - T$, p is on the right of $l_{j+1} - 2\delta$. At the same time $t_{j+1} - T$, q_{j+1} is at most within distance $\frac{\delta T}{T'}$ from l_{j+1}, since it has to be located at l_{j+1} at time t_{j+1}. Then, since $3\delta < \delta_2$, at time $t_{j+1} - T$, p and q_{j+1} are neighbors. p and q_{j+1} remain neighbors throughout all the period $[t_{j+1} - T, t_{j+1}]$ because at time t_{j+1}, p is at most within distance 3δ from l_{j+1}, due to Lemma 4(1) and Observation 2.

By the third bullet of Lemma 4, either q_{j+1} receives I at time t_{j+1} because a node q that received the information directly by q_j invoked *broadcast*$([I, \cdot])$ at some time $\tau \in [t_{j+1} - T, t_{j+1}]$ or q_{j+1} invoked *broadcast*$([I, \cdot])$ also at time $t_{j+1} - T$. In this last case, the claim holds because p and q_{j+1} are neighbors throughout all the period $[t_{j+1} - T, t_{j+1}]$ and because of Observation 3. Then, consider the case where q_{j+1} receives I at time t_{j+1} because a node q invoked *broadcast*$([I, \cdot])$ at some time $\tau \in [t_{j+1} - T, t_{j+1}]$.

If at time $t_{j+1}+T$ node p is within distance Δ from l_{j+1} then by Lemma 3 p delivers the message by time $t_{j+1}+T$. Otherwise, the location of p at time $t_{j+1} + T$ is on the left of $l_{j+1} - \Delta$. This implies that the location of p at time t_{j+1} is minor or equal to $l_{j+1} + \Delta - \frac{\delta T}{T'}$. Then, at time t_{j+1}, p and q are neighbors because distance(q, q_{j+1}, t_{j+1})$< \delta_2$ and distance(p, q_{j+1}, t_{j+1})$\geq \Delta - \frac{\delta T}{T'}$.

Note that at time t_j, p is on the right of l_j. Then, by Observation 4 either p delivers the information by time $t_j + T$ or at some point $t \in [t_j, t_j + T]$ p is located on the right of q. Note that q will broadcast the information once in each time interval $[t_j + kT, t_j + (k+1)T]$ with $k \in \{0, \dots, 3\}$. So either there is a time in $[t_j, t_{j+1}]$ where p and q are strong neighbors and then p delivers the information by time $t_{j+1} + T$, or at time t_{j+1} q is on the left of p and this latter is on the left of q_{j+1}. Then, p will deliver information I by time $t_{j+1} + 2T$ because of a call of broadcast either at q_{j+1} or at q. This is because either p remains neighbors of q or of q_{j+1} throughout all the interval $[t_{j+1} - T, t_{j+1} + T]$ or at time t_{j+1}, p and q are within a distance greater than δ_1 from each other and they move towards or in the same direction of q. So they do not disconnect for at least other $2T$. □

Lemma 8. *Let p be a node that at some time $t \in [t_j, t_{j+1}]$ is in some location $l_p \in [l_j, l_{j+1}]$. If there does not exist a time $t' \in [t_j, t_{j+1}]$ with $t' > t$ such that p is not on the right of l_{j+1}, p delivers the information I by time $t_{j+1} + 2T$.*

Proof. Let p be a node that at some time $t \in [t_j, t_{j+1}]$ is located at $l_p \in [l_j, l_{j+1}]$. Assume that it does not exist a time $t' \in [t_j, t_{j+1}]$ with $t' > t$ such that p is not on the right of l_{j+1}. Then if at time t_j p is either on the left of l_j or on the right of l_{j+1}, then the claim follows respectively by Lemma 5 and Observation 3 and by Lemma 7. Finally, consider the case where node p is inside the interval $[l_j, l_{j+1}]$ throughout all the interval $[t_j, t_{j+1}]$. We prove that p delivers information I by time $t_{j+1} + T$. If at time $t_j + T$ p is within distance Δ from l_j, p delivers the information I by time $t_j + T$, because of Lemma 9. Then assume that p is located in the interval $[l_j + \Delta, l_{j+1}]$ at time $t_j + T$. At that time q is located on the left of location $l_j + \Delta$. At time $[t_{j+1} - T, t_{j+1}]$, q and q_{j+1} are neighbors because of the third bullet of Lemma 4. At time $t_{j+1} - T$ one of the following cases will happens: (1) p is in between q and q_{j+1}, (2) p is on the right of both these nodes but on the left of l_{j+1} or (3) p is on the left of both q and q_{j+1}. But this means that p is a neighbor of q throughout $[t_{j+1}, t_{j+1} + 2T]$ or is a neighbor of q_{j+1} throughout $[t_{j+1} - T, t_{j+1}]$. Since q broadcasts I once in each time interval $[t_j + kT, t_j + (k+1)T]$ with $k \in \{0, \dots, 3\}$ and q_{j+1} broadcasts at time t_{j+1}, p delivers I by time $t_{j+1} + 2T$ and the claim holds. □

Now we prove that if a node stays within distance d from the location where the geocast has been invoked, throughout all the geocast period, then it is eventually inside one of the intervals between two consecutive broadcasts at the right time and for long enough to deliver the information I.

Lemma 9. *If a node q stays within distance d from l throughout $[t_0, t_{i+1}]$ for i such that $l + d \in [l_i, l_{i+1}]$, then q delivers the information I by time $t_{i+1} + 2T$.*

Proof. Let t_0 be the time when the source node s performs the first *broadcast*$([I, \cdot])$ because of a call of *M-Geocast*(I, d). If q is located at $l_0 (= l)$ at time t_0 then the lemma holds. Otherwise, without loss of generality, let q be located on the right of s at time t_0. For every time in $[t_0, t_{i+1}]$, q is located either on or on the left of l_{i+1} because $l + d \leq l_{i+1}$.

By induction on j, it is easy to see that there exists a $j \leq i$ such that at time $t \in [t_j, t_{j+1}]$, q is in the interval $[l_j, l_{j+1}]$ and there does not exist a time $t' \in [t_j, t_{j+1}]$ with $t' > t$ such that q is on the right of l_{j+1}. Otherwise at time t_{j+1}, q is on the right of l_{j+1}, and for $j = i$ we have that at time t_{i+1}, q is on the right of l_{i+1}. This means that at time t_{i+1}, q is at a distance greater than d from l. By the Lemma 8, q will deliver the information I by time $t_{i+1} + 2T$. □

Observation 5. *Let $count_I$ be the counter associated to the communication generated by a call of M-Geocast(I, d). $count_I$ is set to zero once when the source invokes the first broadcast$([I, \cdot])$ at time t_0 and it is never reset.*

Observation 6. *Let p be a node different from the source node. p invokes broadcast$([I, \cdot])$ at some time t only if it has generated a receive$([I, \cdot])$ event at some time before t.*

Lemma 10. *Let t be the time when a call of M-Geocast(I, d) is invoked. Every message broadcast or received at some time in $[t, t + k]$ has a counter at most equal to k.*

Proof. The proof is by induction on k. For $k = 0$, we have to consider the time t. At that time only the source node invokes *broadcast*$([I, \cdot])$ and the counter of the broadcast message has a value 0 (line of Fig. 6). Then the claim holds. By inductive hypothesis, assume that every message broadcast or received at some time in $[t, t + k]$ has a counter at most equal to k. Then we prove that every message broadcast or received at some time in $[t, t + k + 1]$ has a counter at most equal to $k + 1$. We know that this cannot happen by time $t + k$ because of the inductive hypothesis. Then by contradiction assume that there exists a message that is received at time $t + k + 1$ and whose counter has value greater than $k + 1$. But since it takes at least 1 time unit to receive a message, this means that the message received at time $t + k + 1$ was broadcast at the latest at time $t + k$. But then if the message has a counter $k + 1$ we contradict the inductive hypotheses.

Finally, consider the case where at time $t + k + 1$ a message m is broadcast by a node p. p increments its counter possibly each time it receives a message or when it broadcast a message. But by time $t + k$ all the messages received by p have a counter smaller or equal to k and p may has broadcast at most k messages. So at time $t + k$ the counter of p is at most k. Then, when at time $t + k + 1$ it broadcasts a message, this message has a counter at most $k + 1$. Then the claim follows. □

Finally we define the bound for the time to ensure the reliable delivery property and the termination property. From this latter, we obtain the bound for the integrity property.

Theorem 10. *If strong connectivity holds, $T' > T$, and $\delta_1 > \delta$, the M-Geocast(I, d) algorithm with $M = 3T(i + 1) + 2T$ and $i = \lfloor \frac{d}{\delta_1 - \frac{\delta T}{T'}} \rfloor$ ensures*

(1) *the reliable delivery Property 1 for $C = 3T(i + 1) + 2T$,*
(2) *the termination Property 2 for $C' = (3T(i + 1) + 2T + 1)T$, and*
(3) *the integrity Property 3 for $d' = (C' + T)(\delta_2 + \frac{\delta}{T'})$.*

Proof. Let us first prove (1). From Lemma 4, we know that any $3T$ rounds starting from $t_0 = t$ the information reaches some distance $\delta_1 - \frac{\delta T}{T'}$ further from l. Formally, $l_i - l \geq i(\delta_1 - \delta T/T')$. Since we want that all the nodes during the geocast interval remain within distance d from l deliver the information I, we need to compute the maximum value that i could take in any execution such that $(l + d) \in [l_i, l_{i+1})$. Then $i \leq \lfloor \frac{l_i - l}{\delta_1 - \frac{\delta T}{T'}} \rfloor$ and because $l_i - l \leq d$, $i \leq \lfloor \frac{d}{\delta_1 - \frac{\delta T}{T'}} \rfloor$.

From Lemma 9, all the nodes that remain within distance d from $l (= l_0)$ throughout $[t_0, t_{i+1}]$ deliver I by time $t_{i+1} + T = t + C$. By Lemma 4, $t_{i+1} - t \leq 3T(i + 1)$, and $C = t_{i+1} - t + T \leq 3T(i + 1) + 2T$. Then $C \leq 3T(\lfloor \frac{d}{\delta_1 - \frac{\delta T}{T'}} \rfloor + 1) + 2T$.

We have to finally prove that during $[t, t + C]$, for any node $count_I < M$ where $M = C$. This follows by Lemma 10.

We now prove (2). Every message received causes rebroadcasting of I in a message with a counter at least incremented by one, and this will happen at least once every T times. Termination happens after any message received has a counter larger than $3T(i+1)+2T$, where $i = \lfloor \frac{d}{\delta_1 - \frac{\delta T}{T'}} \rfloor$. This happens within $(3T(i+1)+2T+1)T+T$ times, because all messages broadcast after time $(3T(i + 1) + 2T + 1)T$ have counters at least equal to $3T(i + 1) + 2T + 1$ and all such messages are received within at most another T times. Note that in the worst case each broadcast message is received exactly after T times and then the counter counts one time more while in reality T times are the last. Therefore, $C' = (3T(\lfloor \frac{d}{\delta_1 - \frac{\delta T}{T'}} \rfloor + 1) + 2T + 1)T$.

Finally, we prove (3). A broadcast message will be received at least after one time unit during which any node can traverse distance at most $\frac{\delta}{T'}$. Therefore, if a node broadcasts a message from location l' at time t', then its neighbors receive it the earliest at time $t' + 1$, when at a distance less than $\delta_2 + \frac{\delta}{T'}$ away from l'. Then, if the source starts *M-Geocast*(I, d) at time t from location l, at time $t + m$, the furthest node that delivers I is at a distance less than $m(\delta_2 + \frac{\delta}{T'})$ away from l. By (2), after time $t + C'$, no node broadcasts messages with information I. Therefore, no node delivers I after time $t + C' + T$. But at time $t + C' + T$, all nodes that have delivered I are within a distance less than $(C' + T)(\delta_2 + \frac{\delta}{T'})$ from l. Therefore, if a node remains further than $d' = (C' + T)(\delta_2 + \frac{\delta}{T'})$ from l, it will never deliver I. □

6.2. Algorithm for the two-dimensional case

From Corollary 1 we know that it is impossible to solve the geocast problem considering an infinite set of nodes, when nodes move on a plane. Therefore, we need to assume a finite set of nodes to provide the solution. This is still challenging because it is hard to estimate bounds on the hop distance based on the physical distance of the nodes. Note that although such bounds were not explicitly calculated in the one-dimensional case, the correlation of hop distance and physical distance between nodes was the key in proving that information propagates away from its original location and eventually it arrives to all nodes within the appropriate area. In the two-dimensional case, we cannot make similar arguments because the physical distance between nodes does not give any information about their hop distance. For example, consider two non-neighboring nodes p and q that are located very close to each other. Even if those do not move (causing their physical distance to be fixed), their hop distance can change dramatically and the path linking them could contain nodes located very far from p and q. (A scenario like this one was used to prove Theorem 9.) If p initiates a call of Geocast(I, d) and q is within distance d from p, I will have to travel through the (dynamically changing) long path that connects them (to ensure reliable delivery). Hence, the terminating time of the algorithm no longer depends solely on the physical distance d but also on the hop distance between p and q which is hard (if possible) to calculate in our model. To deal with the challenges of the two-dimensional case, we additionally assume a known upper bound n on the number of nodes. This additional information provides an upper bound on the hop distance of any two nodes. As we show next, this suffices for a natural geocast algorithm to work in two dimensions.

We show now that the algorithm *M-Geocast*(I, d) solves the geocast problem in two dimensions for an appropriate value of M. Let us denote by S the set of nodes that have already delivered the information I, and $S(t)$ the set S at time t. Let us denote by t_i the time at which the set S increases from size i to $i + 1$. Note that t_0 is the time the geocast starts.

Lemma 11. *If $T' > T$ and $count_I < M$ at all nodes during the time interval $[t_0, t_{n-1}]$, then $t_{i+1} - t_i \leq 3T$ for every $i \in \{0, \dots, n-2\}$.*

Proof. Since strong connectivity holds, at any time there must be chains of strong neighbors connecting any two nodes in the system. In particular, at every time $t_0 < t < t_{n-1}$ (i.e., such that $S(t) \neq \Pi$) there must exist at least a pair of neighbors q and p such that $q \in S(t)$ and $p \notin S(t)$. Let $C(t)$ denote the set of all such pairs.

Let us fix an $i \in \{0, \dots, n-2\}$, and assume, for contradiction, that $t_{i+1} - t_i > 3T$. Consider the case when there is some pair $(q, p) \in C(t_i)$ that belongs to $C(t')$ for all $t' \in [t_i, t_i + 2T]$. In other words, this pair is formed by a node q that has I at t_i, and a node p that does not, neighbors for at least $2T$ time. By the *M-Geocast*(I, d) algorithm and the fact that $count_I < M$ during the time interval $[t_0, t_{n-1}]$, a node having the information I will rebroadcast it once every T time. Hence q will rebroadcast the information I at some time $t' \in [t_i, t_i+T]$, and thus p will receive and deliver it by time $t'+T \leq t_i+2T$.

Otherwise, all the connections in $C(t_i), i \in \{0, \dots, n-2\}$, have been broken by some time $t' \in (t_i, t_i+2T]$. But, for strong connectivity to hold, a strong connection has to exist between some node $q \in S(t')$ and a node $p \notin S(t')$, since otherwise these subsets are disconnected at time t'. Let $t'', t_i < t'' \leq t'$, be the time at which q and p become strong neighbors, i.e. they are within distance δ_1 from each other. The claim follows if $q \notin S(t_i)$, since $q \in S(t')$ and $t_i < t'' \leq t' \leq t_i + 2T$. Otherwise, note that by Lemma 1 and the fact that $T' > T$, q and p are neighbors throughout all the period $[t'' - T, t'' + T]$. Then, since $q \in S(t_i)$ and $t_i < t''$, q will broadcast I once in the period $[t'' - T, t'']$, and p will deliver I by time $t'' + T > t' > t_i$. Given that $t'' \leq t_i + 2T$, p will deliver the information I by time $t_i + 3T$ and the claim holds. □

Let us now relate the value of the $count_I$ at each node with respect to the time that has passed since *M-Geocast*(I, d) was invoked. Let $count_I(q, t)$ be the value of the variable $count_I$ of node q at time t. Let us define a *propagation sequence* as the sequence of nodes $s = p_0, p_1, p_2, \dots, p_k = q$ such that the first message received by p_i with information I was sent by p_{i-1}. Node $s = p_0$ is the source of the geocast.

Lemma 12. *Let t_0 be the time at which M-Geocast(I, d) is invoked at source s. Given a node q with propagation sequence $s = p_0, p_1, p_2, \dots, p_k = q$ and a time $t \geq t_0$ at which q has delivered I, with $count_I(q, t) \leq M$, then it is satisfied that $((t - t_0) - count_I(q, t)) \in [0, k(T - 1) + T]$.*

Proof. We prove by induction on k that at time $t \geq t_0$ it is satisfied that $((t - t_0) - count_I(p_k, t)) \in [0, k(T - 1) + T]$, and that if a message is sent at time t it carries a counter $c(p_k, t)$ such that $((t - t_0) - c(p_k, t)) \in [0, k(T - 1)]$. The base case is the source node $s = p_0$. At time t_0 the source sets $count_I(s, t_0) = 0$ (Line 9), and then, as long as $count_I < M$, it increments $count_I$ by T every T time (Line 12). Hence, at time $t = t_0 + \alpha$ we have $((t - t_0) - count_I(s, t)) = 0$ if α is a multiple of T, and $((t - t_0) - count_I(s, t)) > 0$ otherwise. Furthermore, the difference $(t - t_0) - count_I$ is always smaller than T. Since messages are broadcast at times $t = t_0 + \alpha$ with α a multiple of T, the values $c(s, t)$ carried by the messages sent by the source satisfy $((t - t_0) - c(s, t)) = 0$.

Let us assume now by induction that, if p_{i-1} broadcasts a message at time $t \geq t_0$, this carries a value $c(p_{i-1}, t)$ such that $((t - t_0) - c(p_{i-1}, t)) \in [0, (i - 1)(T - 1)]$. If p_i receives I for the first time at t' and the corresponding message was sent by p_{i-1} at time t, p_i sets $count_I(p_i, t') = c(p_{i-1}, t) + 1$ (Line 4). This message took between 1 and T time units to be received at time $t' = t + \alpha$. Hence, $\alpha \in [1, T]$. Considering one extreme case, if $((t - t_0) - c(p_{i-1}, t)) = 0$ and $\alpha = 1$, then $((t' - t_0) - count_I(p_i, t')) = 0$. In the other extreme, if $((t - t_0) - c(p_{i-1}, t)) = (i - 1)(T - 1)$ and $\alpha = T$, then $((t' - t_0) - count_I(p_i, t')) = i(T - 1)$. Therefore, $((t' - t_0) - count_I(p_i, t')) \in [0, i(T - 1)]$. Like the source, p_i increments $count_I$ by T every T time as long as $count_I < M$ (Line 12). Hence, at any time $t'' = t' + \alpha$ we have $((t'' - t_0) - count_I(p_i, t'')) \in [0, i(T - 1)]$ if α is a multiple of T. Otherwise, this difference increases in up to T time, and hence $((t'' - t_0) - count_I(p_i, t'')) \in [0, i(T - 1) + T]$. Since messages are broadcast by p_i at times $t'' = t' + \alpha$ with α a multiple of T, the values $c(p_i, t'')$ carried by the messages sent by p_i satisfy $((t'' - t_0) - c(p_i, t'')) \in [0, i(T - 1)]$. □

This lemma can be used to prove the following theorem, which shows that the geocast problem can be solved in two dimensions as long as $T' > T$.

Theorem 11. *If strong connectivity holds and $T' > T$, the M-Geocast(I, d) algorithm with $M = 3T(n - 1)$ ensures*

(1) *the reliable delivery Property 1 for* $C = 3T(n-1)$,
(2) *the termination Property 2 for* $C' = 3T(n-1) + (n-1)(T-1) + T$, *and*
(3) *the integrity Property 3 for* $d' = \max(d, 3T(n-1)v_{max} + (n-1)\delta_2)$.

Proof. The first part of the claim is a direct consequence of Lemma 11, which proves that at most $3T(n-1) \geq t_{n-1} - t_0$ time after *Geocast*(I, d) is invoked, all nodes have delivered the information I. The second part of the claim follows from Lemma 12, using the fact that no propagation sequence has more than n nodes (hence taking $k = n-1$), combined with the first part of the claim. The third claim is also a direct consequence of Lemma 11, since the information can be carried by nodes at most distance $3T(n-1)v_{max}$ in time $3T(n-1)$ from the initial location of the source, and travels less than $(n-1)\delta_2$ in the $n-1$ broadcasts that inform new nodes. □

7. Related work

Geocast was introduced by Navas et al. [4,3]. Geocast algorithms for mobile ad-hoc networks [5,10,7,6], unlike our deterministic solution, only provide probabilistic guarantees. This may not suffice. For example, Dolev et al. [8] need a deterministic geocast to implement atomic memory. Deterministic solutions are given for multicast [14,16,17] and broadcast [11] for mobile ad-hoc networks. Both solutions in [14,16] consider a finite and fixed number of mobile nodes arranged somehow in logical or physical structures. They divide the nodes into groups each of which has a special node which coordinates message propagation and collects acknowledgments. Moreover, they make the following stronger than necessary assumption: they require that the network topology stabilizes for periods long enough to ensure delivery. Finally, simulation results [18] show that the approach proposed in [14] does not work if nodes move fast. Bounds that allow the algorithms to work correctly are not presented. Chandra et al. [17] provide a broadcasting algorithm and show by experiments that either all or none of the nodes get the message with high probability. Mohsin et al. [11] implement (deterministic) broadcast for a synchronous mobile ad-hoc network with restricted movement patterns. In particular, nodes move on top of a grid such that at the beginning of each round nodes are located at grid points. They assume that all nodes move at the same constant speed and direction of movement cannot change within a round. Finally, nodes need to inform their neighbors about their future moving pattern for short future time periods.

Some theoretical results [12,13], in the last few years have made an effort to model the mobility of nodes and to understand the impact of mobility on the solvability and the complexity of basic problems (e.g. broadcast, routing) in mobile ad-hoc networks. In particular, [12,13] model the communication topology as a dynamic graph and mobility as an adversary able to change the edges of the graph accordingly to a given strategy. On the contrary, we use a different approach, we define the dynamics of the communication topology in a parametric way in terms of connectivity and speed of movement of nodes.

Few bounds on deterministic communication in MANETs have been provided [15,1]. We prove that the lower time complexity bound to complete a geocast in the plane is $\Omega(nT)$. Interestingly, Prakash et al. [15] provide a lower bound of $\Omega(n)$ rounds for the completion time of broadcast in mobile ad hoc networks, where n is the number of nodes in the network. As the authors point out, they consider grid-based networks, but a lower bound proved for this restricted grid mobility model automatically applies to more general mobility models. This latter result improves the $\Omega(D \log n)$ bound provided by Bruschi and Pinto [1], where D is the diameter of the network. These results unveil the fact that, when nodes may move, the dominating factor in the complexity of an algorithm is the number of nodes in the network and not its diameter.

8. Conclusion

In the context of geocasting, this paper has formally shown how the speed of movement of the nodes creates uncertainty in a distributed system in terms of the cost and solvability of the problem. To the best of our knowledge, this is in fact the first time in which bounds are formally defined on the speed of node movement which make it possible to solve geocasting and relate its time complexity to the speed. This formally verifies the intuition that the faster the nodes move, the more costly it would be to solve geocasting. Moreover, we have shown that when nodes move at speeds higher than a certain bound, geocasting cannot be solved. In particular, for the two-dimensional mobility model, we have presented a tight bound on the maximum speed of movement that keeps the solvability of geocast. We have also proved that $\Omega(nT)$ is a time complex lower bound for a geocast algorithm to ensure deterministic reliable delivery, and we have provided a distributed solution which is proved to be asymptotically optimal in time. In fact, our solution and bounds are also applicable to 3 dimensions, a case that is rarely studied but may be of growing interest.

Assuming the one dimension mobility model, i.e. nodes move on a line, we have proved that $v_{max} < \frac{2\delta}{T}$ is a necessary condition to solve the geocast, where δ is a system parameter, and presented an efficient algorithm when $v_{max} < \frac{\delta}{T}$. This leaves a gap on the maximum speed to solve the geocast problem in one dimension. Interestingly, the time complexity of the algorithm in one dimension depends on the speed of movement of nodes and on the distance d to be covered, while in two dimensions it is related to the size of the system. This is mainly due to the fact that despite mobility, having nodes located in a plane may let the information propagate far away from the source node but in a different direction w.r.t. the nodes that have to be covered. To prove our results we do not make any assumption on the network topology except that it is strongly connected. Whether more efficient solutions exist for specific topologies and according to the speed of movement (which defines how fast the topology may change) is an open problem.

Finally, in our opinion, the importance of the results shown in this paper lies also in the way the impact of the speed of nodes has been studied with respect to problem solvability and the cost of the solution. We believe that the set of steps we followed (i.e., the model, the way solvability problem has been tackled, and how the tradeoff bounds on the cost of solvability has been established) can be a general canvas used to analyze the uncertainty due to speed of nodes introduced within any distributed computing related problem on a mobile setting also considering weaker system model than the one addressed in this paper (e.g. including failures and/or nodes moving in a two-dimensional model).

References

[1] D. Bruschi, M. Del Pinto, Lower bounds for the broadcast problem in mobile radio networks, Distributed Computing 10 (3) (1997) 129–135.
[2] B.N. Clark, C.J. Colbourn, D.S. Johnson, Unit disk graphs, Discrete Mathematics 86 (1–3) (1990) 165–177.
[3] T. Imielinski, J. C. Navas, GPS-based geographic addressing, routing, and resource discovery, Communication of the ACM 42 (4) (1999) 86–92.
[4] J.C. Navas, T. Imielinski, Geocast: geographic addressing and routing, in: Proceedings of the 3rd Annual ACM/IEEE International Conference on Mobile Computing and Networking, MobiCom, ACM Press, 1997, pp. 66–76.
[5] Y. Ko, N. H. Vaidya, Geotora: a protocol for geocasting in mobile ad hoc networks, in: Proceedings of the 8th International Conference on Network Protocols, ICNP, IEEE Computer Society, 2000, p. 240.
[6] Y. Ko, N.H. Vaidya, Flooding-based geocasting protocols for mobile ad hoc networks, Mobile Network and Application 7 (6) (2002) 471–480.
[7] W. Liao, Y. Tseng, K. Lo, J. Sheu, Geogrid: a geocasting protocol for mobile ad hoc networks based on grid, Journal of Internet Technology 1 (2) (2001) 23–32.
[8] Shlomi Dolev, Seth Gilbert, Nancy Lynch, Alexander Shvartsman, Jennifer Welch, Geoquorum: implementing atomic memory in ad hoc networks, in: Proceedings of the 17th International Conference on Principles of DIStributed Computing, DISC, 2003, pp. 306–320.
[9] B.S. Chlebus, L. Gasieniec, A. Gibbsons, A. Pelc, W. Rytter, Deterministic broadcasting in ad hoc radio networks, Distributed Computing 15 (1) (2002) 27–38.
[10] J. Boleng, T. Camp, V. Tolety, Mesh-based geocast routing protocols in an ad hoc network, in: Proceedings of the 15th International Parallel & Distributed Processing Symposium, IPDPS, April 2001, pp. 184–193.
[11] M. Mohsin, D. Cavin, Y. Sasson, R. Prakash, A. Schiper, Reliable broadcast in wireless mobile ad hoc networks, in: Proceedings of the 39th Hawaii International Conference on System Sciences, HICSS, IEEE Computer Society, 2006, p. 233.1.
[12] A.E.F. Clementi, F. Pasquale, A. Monti, R. Silvestri, Communication in dynamic radio networks, in: Proceedings of the 26th Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing, PODC 2007, 2007, pp. 205–214.
[13] R. O'Dell, R. Wattenhofer, Information dissemination in highly dynamic graphs, in: Proceedings of DIALM-POMC 2005, 2005, pp. 104–110.
[14] E. Pagani, G. P. Rossi, Reliable broadcast in mobile multihop packet networks, in: Proceedings of the 3rd Annual ACM/IEEE International Conference on Mobile Computing and Networking, MobiCom, ACM Press, 1997, pp. 34–42.
[15] R. Prakash, A. Schiper, M. Mohsin, D. Cavin, Y. Sasson, A lower bound for broadcasting in mobile ad hoc networks, Ecole Polytechnique Federale de Lausanne, Tech. Rep. IC/2004/37, 2004.
[16] S.K.S. Gupta, P.K. Srimani, An adaptive protocol for reliable multicast in mobile multi-hop radio networks, in: Proceedings of the 2nd Workshop on Mobile Computing Systems and Applications, WMCSA, IEEE Computer Society, 1999, p. 111.
[17] R. Chandra, V. Ramasubramanian, K.P. Birman, Anonymous gossip: Improving multicast reliability in mobile ad-hoc networks, in: Proceedings of the 21st International Conference on Distributed Computing Systems, ICDCS, IEEE Computer Society, 2001, pp. 275–283.
[18] E. Pagani, G.P. Rossi, Providing reliable and fault tolerant broadcast delivery in mobile ad-hoc networks, Mobile Networks and Applications 4 (3) (1999) 175–192.