Mobility versus the Cost of Geocasting in Mobile Ad-Hoc Networks ⋆

Roberto Baldoni, Kleoni Ioannidou, and Alessia Milani

Universitá di Roma *La Sapienza*, Italy
baldoni@dis.uniroma1.it|ioannidu@cs.toronto.edu|
milani@dis.uniroma1.it

Abstract. 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 despite connectivity and no matter how slowly the nodes move. Even if each node maintains a stable connection with each of its neighbours for some period of time, it is impossible to solve geocast if nodes move too fast. Additionally, we give a trade-off lower bound which shows that the faster the nodes can move, the more costly it would be to solve the geocast problem. Finally, for the one-dimensional case of the mobile ad-hoc network, we provide an algorithm for geocasting and we prove its correctness given exact bounds on the speed of movement.
Keywords: Mobile ad-hoc networks, geocast, speed of movement vs cost of the solution, distributed systems.

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 ”geo” related problems in mobile ad-hoc networks such as georouting [1, 2], geocasting [3–6], geoquorums [6], etc.

Geocasting is a variant of the multicast problem [7]. In geocasting, the nodes eligible to deliver a message are the ones that belong to a specific geographical area. Specifications to this problem can be either best effort or deterministic. An

⋆ The work described in this paper was partially supported by the Italian Ministry of Education, University, and Research (MIUR) under the ISMANET project and by the European Community under Resist Network of Excellence.

implementation of a best effort specification aims to maximize the probability that nodes eligible to deliver the information, they actually deliver it [3–5]. Deterministic specifications define a set of nodes and an implementation of such a specification ensures that each of these nodes will deliver the information [6].

When geocasting is solved for mobile ad-hoc networks, the speed of how nodes move becomes an important factor. 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 too high. In an extreme (unrealistic) scenario, nodes can move fast enough to ensure that no two neighbours 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 on geocasting based on deterministic specifications investigating the relation between cost of solving geocasting and mobility. In particular, we firstly 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 makes a distinction between strong and weak connectivity. A system strongly connected has some assurance of topological stability, i.e., there is always a path between every two nodes formed by strong neighbors, where strong neighbors means that they remain neighbors for some period of time. A connected system that does not satisfy the previous property is weakly connected. Our model does not rely on either GPS or synchrony being thus very weak with respect to other models presented in the literature [8]. The geocasting specification is split in 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.

Hence, a general framework of geocasting algorithms is proposed (Section 3.2), which captures existing geocast algorithms. An algorithm belonging to this framework acts as follows: once a node receives a message (with the geocast information) broadcast by a neighbour, it may repeat a (local) broadcast k times, once every α rounds, depending on some condition. Using this framework in our model several results have been proved: (i) if nodes are weakly connected geocasting cannot be solved no matter how slowly the nodes can move (Theorem 1); (ii) if stronger connectivity holds, then geocasting is still impossible for some bound of node's speed of movement (Theorems 2 and 3); (iii) a tradeoff lower bound that relates the cost of geocasting to the speed of movement of nodes (Theorem 4).

Finally, if the speed is small enough, we show how to solve the geocasting problem in a one-dimensional setting (Section 5). We prove that the time complexity of this algorithm increases with the speed of nodes. The algorithm does not require any knowledge of the topology of the system. These results confirm the intuition that the fastest the nodes move, the more expensive it would be to solve the geocasting problem and if nodes move too fast then no solution can be achieved.

2 A Model for Mobile Ad-Hoc Networks

We consider a system of (mobile) nodes which move with bounded speeds in a continuous manner on the plane. There is no known upper bound on the number of nodes in the system and nodes do not fail. Nodes communicate by exchanging messages over a wireless radio network. To define neighbourhood of nodes, let distance(p, p', t) denote the physical distance between two nodes p and p' at time t. Two nodes p and p' are *neighbours* at some time t, if distance(p, p', t)$< r$, for fixed $r > 0$. We assume that each node can have at most H neighbours at each time.

Nodes do not have access to a global clock, instead they have (not necessarily synchronized) local clocks which run at the same rate. Within a small time period, called a *round*, a node can execute in a sequential and atomic manner receiving at most H messages, broadcasting at most one message, and local computation. To perform a local broadcast of a message m, a node p is provided with a primitive denoted *broadcast*(m). It takes at least one round for a broadcast message m to be received by a node which then generates a *receive*(m) event. For simplicity of presentation, the duration of a round is one time unit (i.e., in $[t, t+i]$, i rounds have elapsed). If *broadcast*(m) is performed by node p at time t then all nodes that remain neighbours of p throughout $[t, t+T]$ receive m by time $t+T$, for some fixed integer $T > 0$. It is possible that some nodes that are neighbours of p at times in $[t, t+T]$ also receive m but no node receives m after time $t+T$. If two or more nodes perform broadcasts concurrently there may be interference and messages may be lost. We assume this to be dealt by a lower level communication layer [9] within the T rounds it takes for a message to be (reliably) delivered to its destination. There is no other way that messages can be lost.

Connectivity. The standard definition of connectivity, called *weak connectivity*, ensures that for every pair of nodes p and p' and every time t, there is at least one path of neighbours connecting p and p' at time t. Weak connectivity allows an adversary to continually change the neighbourhood of nodes and render impossible even the basic task of geocasting (Theorem 1). For this reason, we assume a stronger version, called *strong connectivity*. To define this, first, we introduce the notion of strong neighbours. If there is an upper bound on the speed of nodes, then the closer two neighbours are located to each other, the longer they will remain neighbours. Hence, if nodes are located fairly close, then their connection is guaranteed for some period of time. Formally,

Definition 1 (Strong Neighbours). *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 neighbours *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 $t'' \in [t', t]$.*

Assumption 1 (Strong Connectivity) *For every pair of nodes p and p' and every time t, there is at least one path of strong neighbours connecting p and p' at t.*

Two nodes *strongly connect* when they become strong neighbours and they *lose their connection or disconnect* when they cease being neighbours. By increasing δ_1, the set of strong neighbours of each node either remains the same or increases. This is desirable, because then strong connectivity is not too much stronger than weak connectivity. Therefore, for practical applications, we would like to design algorithms considering values of δ_1 that are as large as possible. Because of this, in this paper, we assume $\delta_1 \geq \frac{\delta_2}{2}$.

Mobility. We assume an upper bound on the speed of node movement which exists in practical situations. Then, Lemma 1 describes some topological stability. Formally,

Assumption 2 (Movement Speed) *It takes at least $T' > 0$ rounds for a node to travel distance $\delta = \frac{\delta_2 - \delta_1}{2}$ on the plane.*

From Definition 1 and Assumption 2, we gain some topological stability in the network, which is formally expressed in the following lemma:

Lemma 1. *If two nodes become strong neighbours at time t, then they remain (strong) neighbours throughout $[t, t + T']$ (i.e., for T' rounds).*

Proof. If p and p' become strong neighbours at time t, then distance(p, p', t)$= \delta_1$. To disconnect, they must move away from each other so that their distance is larger than or equal to δ_2 (traversing in total distance at least 2δ). From assumption 2, this takes at least T' rounds when they travel in opposite directions.

3 The Geocast Problem

The goal of geocasting is to deliver information to nodes in a specific geographical area. The geocast problem can be solved by a geocast service, implemented by a geocast algorithm which runs on mobile nodes. The geocast service supports each mobile node with two primitives: Geocast(I, d) to geocast information I at distance d and Deliver(I) to deliver information I. As illustrated in Figure 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 neighbours.

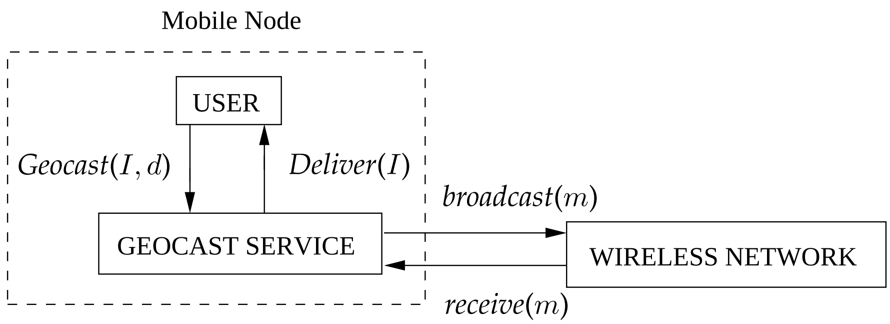

Fig. 1. System Architecture.

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 by all nodes that are located at distance at most d away from l throughout $[t, t + C]$.

The following properties rule out solutions which waste resources causing continuous communication or distribution of information I among all nodes.

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

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

3.2 A General Framework for Geocasting Algorithms

We present a framework, (k, α)-Geocast(I, d) for $\alpha \geq 1$, that describes a large class of geocast algorithms. When the source invokes (k, α)-Geocast(I, d), k messages containing I are broadcast, once every α rounds. When a node receives a message containing I, k broadcasts of messages containing I are generated, once every α rounds as long as some condition (described by a boolean function *CHECK*) holds. *CHECK* can be different for each algorithm in this class.

More precisely, for each call of Geocast(I, d), each node p stores a variable $time_{p,I}$, and a boolean variable $flag_{p,I}$, initially set to $\perp$ and 0, respectively. We denote $clock_p$ the current value of the physical clock at p.

When source s executes (k, α)-Geocast(I, d), $time_{s,I}$ is set to clock$_s$ and every α rounds $flag_{s,I}$ is set to 1 (illustrated in Figure 2 (a)). This causes the broadcast of a message m containing information I (illustrated in Figure 2 (b)).

$(k, \alpha) - Geocast(I, d)$ **by** s
$time_{s,I} \leftarrow clock_s;$
for $(i = 1; i{+}{+}; i \leq k)$
  when $(clock_s == time_{s,I} + [(i-1)\alpha)])$
    $flag_{s,I} \leftarrow 1;$

(a)

WHEN $(flag_{s,I} == 1)$
$trigger\langle broadcast(m)\rangle;$ % $I \in m$ %
$flag_{s,I} \leftarrow 0;$

(b)

Fig. 2. $(k, \alpha)-$Geocast(I, d) algorithm performed by the source s.

The first time a node p executes *receive*(m), $time_{p,I}$ gets the value of $clock_p$. Any time p receives a message with information I, if *CHECK* is true, it sets $flag_{p,I}$ to 1, every α rounds for k times (illustrated in Figure 3 (a)), which in turn causes a broadcast of a message containing I (illustrated in Figure 3 (b)). After each such broadcast, $flag_{p,I}$ is set to 0.

UPON EVENT $\langle receive(m)\rangle$ **by** p
$trigger\langle Deliver(I)\rangle;$ % *I is contained in m* %
$t_{p,I} \leftarrow clock_p;$
if $(time_{p,I} == \bot)$
  then $time_{p,I} \leftarrow t_{p,I};$
if $(CHECK)$
  then for $(i = 1; i{+}{+}; i \leq k)$
    when $(clock_p == time_{p,I} + [\lceil \frac{t_{p,I} - time_{p,I}}{\alpha} \rceil + (i-1)]\alpha)$
      $flag_{p,I} \leftarrow 1;$

(a)

WHEN $(flag_{p,I} == 1)$ **by** p
$trigger\langle broadcast(m)\rangle;$ % $I \in m$ %
$flag_{p,I} \leftarrow 0;$

(b)

Fig. 3. $(k, \alpha)-$Geocast(I, d) algorithm performed by node p.

Note that if p receives more than one message containing I within α rounds, only one broadcast is triggered. Hence at most one broadcast happens at p every α rounds. The above is ensured by setting $flag_{p,I}$ to 1 only at certain times as shown in line 7 of Figure 3 (a). In particular, $flag_{p,I}$ is set to 1, k times, starting at the closest time after $t_{p,I}$ that is equal to $time_{p,I} + j\alpha$ (where j is an integer).

4 Impossibility Results

End-to-end communication is impossible if the system remains disconnected. Eventual connectivity [10] ensures the existence of a path between sender and receiver with edges which transmit infinitely many messages if infinitely many messages are sent through. Eventual connectivity is necessary for achieving end-to-end communication in general networks. For our mobile ad-hoc network, we show that it is impossible to solve the geocast problem using algorithms in (k, α) - Geocast under weak connectivity, or under strong connectivity if nodes move too fast. To do so, we relate the speed of movement (which is inversely related to T') to the speed of communication (which is inversely related to T). We also show how the speed of nodes relates to the cost of any (k, α) - Geocast algorithm.

For the following impossibility results, we set *CHECK* to true because if reliable delivery is impossible when the maximum broadcasts are allowed, it is also impossible for less broadcasts.

The fact that the (k, α) - Geocast class of algorithms contains a large class of natural geocasting algorithms (including existing ones) makes our impossibility results significant for practical applications. We note that the following lower bounds are not necessarily tight.

Theorem 1. *No algorithm in (k, α) - Geocast(I, d) can solve the geocast problem under the weak connectivity assumption no matter how slowly the nodes move.*

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 neighbour, p, of s is on its right at distance $r - d_\epsilon$ from l, at position l_1 such that $d_\epsilon \leq \frac{v \min\{\alpha,T\}}{2}$. There is a node q located on the right of p at distance d_ϵ from p at position l_2, as illustrated in Figure 4.

Because $d_\epsilon \leq \frac{v \min\{\alpha,T\}}{2}$, distance $2d_\epsilon$ can be traversed during $\min\{\alpha, T\}$ rounds. From state s_{pq} at time t, node q moves with speed $\frac{2d_\epsilon}{\min\{\alpha,T\}}$ until it reaches location l_1 at time $t + \frac{\min\{\alpha,T\}}{2}$. Then, node p moves away from l with speed $\frac{2d_\epsilon}{\min\{\alpha,T\}}$ until it reaches location l_2 at time $t + \min\{\alpha, T\}$. The state, s_{qp}, reached is the same as s_{pq} if we replace p by q and q by p. Weak connectivity is preserved.

Because the switch between s_{pq} and s_{qp} takes $\min\{\alpha, T\}$ rounds, and according to the algorithm at most one broadcast can be initiated every α rounds, it is possible to create an execution where the source starts a local broadcast either at state s_{pq}, or at state s_{qp} and its neighbourhood changes within $\min\{\alpha, T\}$ rounds. This implies that no node ever delivers I. In particular, if $\alpha > T$ then there can be at most one message broadcast every T rounds and this message will be lost because the neighbourhood changes within $\min\{\alpha, T\} = T$ rounds.

Otherwise, $\alpha \leq T$. Since there is at most one message broadcast every α rounds, every such message will be lost because the change in the neighbourhood happens within $\min\{\alpha, T\} = \alpha \leq T$ rounds.

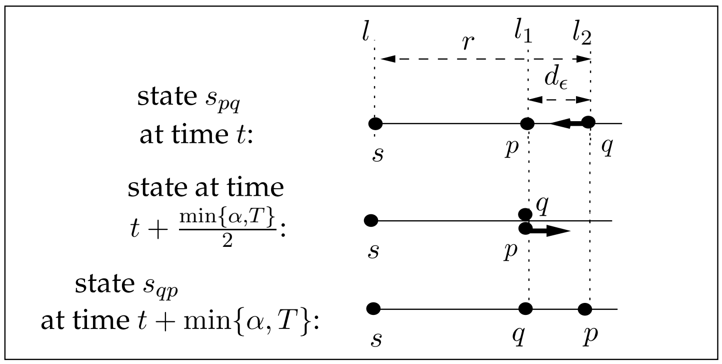

Fig. 4. Proof of Theorem 1.

As stated in Section 2, we assumed that $\delta_1 \geq \frac{\delta_2}{2}$ which is reasonable for practical applications. The following results in this section hold given this assumption. Our lower bounds would be stronger if they held for all values of δ_1. This extension would be of theoretical interest and we leave it as future work.

Theorem 2. *No algorithm in (k, α) - Geocast(I, d) can solve the geocast problem if $T' < \frac{T}{4}$ even if strong connectivity holds.*

Proof. Consider a (k, α)-Geocast(I, d) algorithm executed at time t by the source s. We will describe an execution of this algorithm (illustrated in Figure 5) during which no node (other than s) knows information I, violating reliable delivery. Let s_{pq} be the state at time t with the following properties: all nodes are located on a single line; the source, s, is the leftmost node located at position l; the first node, p, located on the right of l is at position l_1 at distance δ_1 from l; the second node, q, located on the right of l is at position l_2 at distance δ_2 from l and at distance $\delta_2 - \delta_1 = 2\delta$ from l_1; all other nodes of the system are located on the right of s at distance at least $\delta_1 + \delta_2$ from l. Node p is the only (strong) neighbour of s. Nodes s and q are the only (strong) neighbours of p, p is the only (strong) neighbour of q located on the left of q at time t, and the remaining (strong) neighbours, Q, of node q are located on its right at distance exactly δ_1. We conclude that strong connectivity holds at state s_{pq}. Assume that, from state s_{pq}, p moves from l_1 to l_2 and q moves from l_2 to l_1 on a straight line with their highest speed. Each of them will traverse a path of distance 2δ and arrive at its destination at time $t + 2T'$ (by the communication speed assumption). Strong connectivity holds throughout $[t, t + 2T']$ because throughout $[t, t + 2T')$, the sets of strong neighbours of every node in the system does not change, and the state, s_{qp}, reached at time $t + 2T'$ is the same as the state at time t if we replace p by q and q by p. If the above movement happens continually, then for any even integer i, we reach state s_{pq} and for any odd integer i, we reach state s_{qp} at time $t_i = t + 2T'i$.

Let t' be a time at which the source s performs a (local) broadcast during its call of (k, α)-Geocast(I, d). We consider the following two cases for $i =$ odd (the proof for $i =$ even is symmetrical):

- There is i such that $t_i = t'$. Because i is odd, the system is in state s_{qp} at time t_i, it reaches state s_{pq} at time t_{i+1}, and $t_{i+1} - t_i = 2T'$. At time t', q is the only neighbour of s. Node q will stop being a neighbour of s at time $t_{i+1} = t_i + 2T' = t' + 2T'$ which happens before time $t' + T$ because $T' < \frac{T}{4}$. Therefore q will not receive the message being broadcast by s at time t'.
- Otherwise, there is i such that $t_i < t' < t_{i+1}$. Because i is odd, the system is in state s_{qp} at time t_i and the only neighbours of s at time t' are p and q. But node q will cease being a neighbour of s at time t_{i+1} and node p will cease being a neighbour of s at time t_{i+2}. The local delivery of the broadcast message completes at time $t' + T$. Node q will not receive the message broadcast at time t' by s because $t' + T > t_i + 2T' = t_{i+1}$. Similarly, p will not receive this message because $t' + T > t_i + 4T' = t_{i+2}$.

In both cases, any node that will become a neighbour of s after time t' will not receive the broadcast message either.

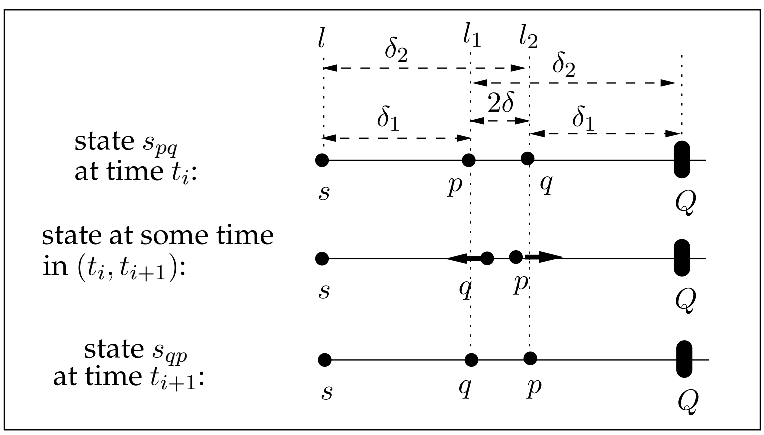

Fig. 5. Proof of Theorem 2.

Theorem 3. *No algorithm in (k, α) - Geocast(I, d) can solve the geocast problem if $T' < \frac{\delta T}{\delta_1}$, for a system with unbounded number of nodes even if strong connectivity holds.*

Proof. We describe an execution (illustrated in Figure 6) 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 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 neighbours 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 neighbours 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 $T' < \frac{T\delta}{\delta_1}$ which implies that $\frac{\delta_1}{T}$ is smaller than the maximum speed (i.e., $\frac{\delta}{T'}$). All other nodes p_i for $i \geq 0$ form a path such that each two consecutive nodes are strong neighbours. Furthermore, q is always a strong neighbour 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 distance $\frac{T\delta_1}{T} = \delta_1$ and it moves towards the left starting from a location at 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 neighbour of any node on or on the left of position l, it will never deliver I.

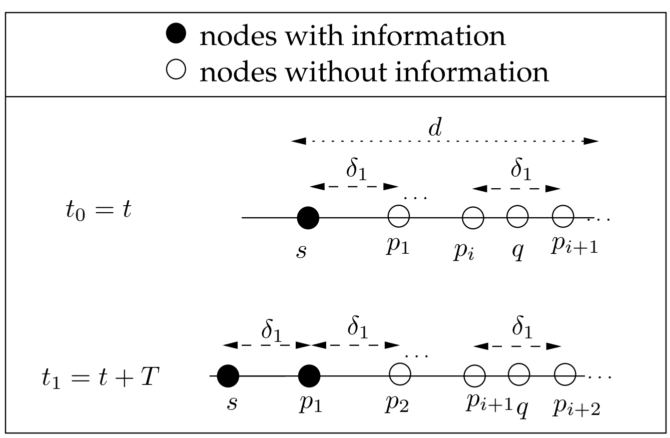

Fig. 6. Proof of Theorem 3.

Theorem 4. *Assuming that $T' > \max\{\frac{1}{4}, \frac{\delta}{\delta_1}\}T$, then if it is possible to solve geocast, 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 (k, α)-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 Figure 7) 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, $s = q_0$. At time t_0, the nodes (other than q) are placed on a line on the right of q_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 neighbours 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 neighbours 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 $i \geq 0$, other than q) form a path of strong neighbours and q is a strong neighbour 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 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 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 neighbours has delivered I. The earliest this happens is within T rounds after I is delivered by a neighbour of q on its left. This neighbour 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., $j > \lfloor \frac{d-\delta_2}{\delta_1 - \frac{T\delta}{T'}} \rfloor$).

Theorem 4 verifies the intuition that the larger the speed of the nodes can be (which is inversely related to T') the more time it would take to solve geocasting.

5 A Geocasting Algorithm

We consider a special case of the mobile ad-hoc model, called *one-dimensional* mobile ad-hoc model, for which the nodes move on a line. Inter-vehicle communication [11] is an application of geocast in this model. For simplicity, the line is straight and horizontal and the locations are real numbers representing points which increase towards the right. We show that $(7, T)$ - Geocast(I, d) works if $T' > 9T$. We attach a counter, $cmsg$, to each message which is set to zero only in the first message broadcast by the source. Each node maintains, in a local counter, the largest counter value it has either received or broadcast. Every time it is ready to broadcast (i.e. its flag is set to 1), it increments its local counter by one and appends this new value to the message. Upon receiving a

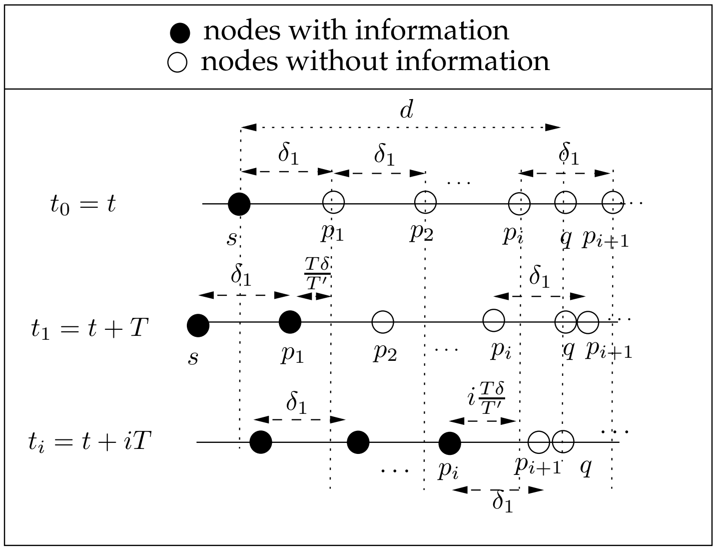

Fig. 7. Proof of Theorem 4.

message with counter $cmsg$, the receiver evaluates *CHECK* which returns true iff $cmsg \leq 6T(i+1) + 2T$, where $i = \lfloor \frac{d}{\delta_1 - \frac{6\delta T}{T'}} \rfloor$. Assume that the source $s = q_0$ initiates a call of $(7,T)$ - 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 (at time t_j and location l_j), to another node, q_{j+1} at some large distance away. The proofs of lemmata 2, 3, and 4 used for correctness appear in the full version of the paper [12].

Lemma 2. *If $T' > 9T$ and node q_j delivers I at time t_j when at location l_j then, assuming that* CHECK *returns true for all nodes throughout $[t_j, t_{j+1}]$, there is a node q_{j+1} which delivers I at time t_{j+1} at location l_{j+1} such that $t_{j+1} - t_j \leq 6T$ and $l_{j+1} - l_j \geq \delta_1 - 6\delta T/T'$.*

Lemma 3. *If $T' > 9T$ and there is a j such that a node p is located in $[l_j, l_{j+1}]$ at some time $t \in [t_j, t_{j+1}]$, then, assuming that* CHECK *returns true for all nodes throughout $[t_j, t_{j+1}]$, p delivers I by time $t_{j+1} + 2T$.*

Lemma 4. *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 there is $j \leq i$ such that q is located at some position in $[l_j, l_{j+1}]$ at some time in $[t_j, t_{j+1}]$.*

Theorem 5. *If $T' \geq 9T$, then $(7,T)$ - Geocast(I,d) ensures reliable delivery for $C \leq 6T(i+1) + 2T$ rounds, where $i = \lfloor \frac{d}{\delta_1 - \frac{6\delta T}{T'}} \rfloor$.*

Proof. During $[t, t+C]$, any node's *CHECK*=true because if geocast starts at time t, then (by induction on time) during $[t, t+C]$, all messages broadcast or received have counters at most equal to C. Then, we show that $C \leq 6T(i+1)+2T$, where $i = \lfloor \frac{d}{\delta_1 - \frac{6\delta T}{T'}} \rfloor$. First, we calculate the maximum value that i could take in any execution such that $l_i \leq l + d$ (i.e., $(l+d) \in [l_i, l_{i+1})$). Next, we show that it suffices that I gets delivered and rebroadcast by a node at location l_{i+1}. From Lemma 2, $l_i - l \geq i(\delta_1 - 6\delta T/T')$. Then $i \leq \lfloor \frac{l_i - l}{\delta_1 - \frac{6\delta T}{T'}} \rfloor$ and because $l_i - l \leq d$, $i \leq \lfloor \frac{d}{\delta_1 - \frac{6\delta T}{T'}} \rfloor$. It remains to calculate C. All nodes that remain within distance d from $l(= l_0)$ throughout $[t, t+C]$, also remain within that distance throughout $[t_0, t_{i+1}]$, (recall that $t = t_0$). If p remains in this area throughout $[t_0, t_{i+1}]$ then from Lemma 4, there is a j such that p is located at some position in $[l_j, l_{j+1}]$ at some time in $[t_j, t_{j+1}]$ for $j \leq i$ and from Lemma 3, p delivers I by time $t_{j+1} + 2T$. Therefore, since $j \leq i$, all nodes within distance d from l deliver I by time $t_{i+1} + 2T = t + C$. By Lemma 2, $t_{i+1} - t \leq 6T(i+1)$ and $C \leq 6T(\lfloor \frac{d}{\delta_1 - \frac{6\delta T}{T'}} \rfloor + 1) + 2T$.

Theorem 6. *If $T' \geq 9T$ then $(7,T)$ - Geocast(I,d) ensures termination for $C' = (6T(i+1) + 2T + 1)T + 8T$ rounds, where $i = \lfloor \frac{d}{\delta_1 - \frac{6\delta T}{T'}} \rfloor$.*

Proof. Every message received causes rebroadcasting of I in a message with counter at least incremented by one and this will happen at least once every T rounds (for at least 7 times). Termination happens within $7T$ rounds from the time after which any message received has counter larger than $6T(i+1)+2T$, where $i = \lfloor \frac{d}{\delta_1 - \frac{6\delta T}{T'}} \rfloor$. This happens within $(6T(i+1) + 2T + 1)T + T$ rounds, because all messages broadcast after time $(6T(i+1) + 2T + 1)T$ have counters at least equal to $6T(i+1) + 2T + 1$ and all such messages are received within at most another T rounds. Therefore, $C' = (6T(\lfloor \frac{d}{\delta_1 - \frac{6\delta T}{T'}} \rfloor + 1) + 2T + 1)T + 8T$ rounds.

Theorem 7. *If $T' \geq 9T$ then $(7,T)$ - Geocast(I,d) ensures integrity.*

Proof. A broadcast message will be received at least after one round 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 neighbours receive it the earliest at time $t'+1$, when at distance less than $\delta_2 + \frac{\delta}{T'}$ away from l'. Then, if the source starts $(7,T)$ - Geocast(I,d) at time t from location l, at time $t+m$, the furthest node that delivers I is at distance less than $m(\delta_2 + \frac{\delta}{T'})$ away from l. By Theorem 6, 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 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 Related Work

Geocast was introduced by Navas et al. [2, 1]. Geocast algorithms for mobile ad-hoc networks [3, 7, 5, 4], unlike our deterministic solution, only provide probabilistic guarantees. This may not suffice. For example, Dolev et al. [6] need deterministic geocast to implement atomic memory. Deterministic solutions are given for multicast [13–15] and broadcast [8] for mobile ad-hoc networks. Both solutions in [13, 14] 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 [16] show that the approach proposed in [13] does not work if nodes move fast. Bounds that allow the algorithms to work correctly are not presented. Chandra et al. [15] 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. [8] 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 neighbours about their future moving pattern for short future time periods.

7 Conclusion and Future Work

To the best of our knowledge, this is 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 that the faster nodes move, the most costly it would be to solve geocasting. Our upper bounds and lower bounds do not match neither for the cost of geocast, nor for the bounds on speed of movement. Although the gap is not large, it would have theoretical interest to match these bounds. We proved our results for the case where $\delta_1 \geq \frac{\delta_2}{2}$. It is unknown whether our lower bounds still hold for $\delta_1 < \frac{\delta_2}{2}$. Another future direction would be to design a geocast algorithm that works for a two-dimensional model including failures.

References

1. T. Imielinski and J. C. Navas. Gps-based geographic addressing, routing, and resource discovery. *Communication of the ACM*, 42(4):86–92, 1999.
2. J.C Navas and Imielinski T. Geocast: geographic addressing and routing. In *Proceedings of the 3rd Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, pages 66–76. ACM Press, 1997.

3. Y. Ko and N. H. Vaidya. Geotora: a protocol for geocasting in mobile ad hoc networks. In *Proceedings of the 8th International Conference on Network Protocols (ICNP)*, page 240. IEEE Computer Society, 2000.
4. Y. Ko and N. H. Vaidya. Flooding-based geocasting protocols for mobile ad hoc networks. *Mobile Network and Application*, 7(6):471–480, 2002.
5. W. Liao, Y. Tseng, K. Lo, and J. Sheu. Geogrid: A geocasting protocol for mobile ad hoc networks based on grid. *Journal of Internet Technology*, 1(2):23–32, 2001.
6. Shlomi Dolev, Seth Gilbert, Nancy Lynch, Alexander Shvartsman, and Jennifer Welch. Geoquorum: Implementing atomic memory in ad hoc networks. In *Proceedings of the 17th International Conference on Principles of DIStributed Computing (DISC)*, pages 306–320, 2003.
7. J. Boleng, T. Camp, and V. Tolety. Mesh-based geocast routing protocols in an ad hoc network. In *Proceedings of the 15th International Parallel & Distributed Processing Symposium (IPDPS)*, pages 184–193, April 2001.
8. M. Mohsin, D. Cavin, Y. Sasson, R. Prakash, and A. Schiper. Reliable broadcast in wireless mobile ad hoc networks. In *Proceedings of the 39th Hawaii International Conference on System Sciences (HICSS)*, page 233.1. IEEE Computer Society, 2006.
9. C. Koo, V. Bhandari, J. Katz, and N. H. Vaidya. Reliable broadcast in radio networks: the bounded collision case. In *Proceedings of the 25th Annual ACM Symposium on Principles of Distributed Computing (PODC)*, pages 258–264, 2006.
10. Faith Ellen. End to end communication. In *Proceedings of the 2nd International Conference On Principles Of Distributed Systems (OPODIS)*, pages 37–44. Hermes, 1998.
11. Abderrahim Benslimane. Optimized dissemination of alarm messages in vehicular ad-hoc networks (vanet). In *Proceedings of the 7th IEEE International Conference on High Speed Networks and Multimedia Communications (HSNMC)*, pages 655–666, 2004.
12. Roberto Baldoni, Kleoni Ioannidou, and Alessia Milani. Mobility versus the cost of geocasting in mobile ad-hoc networks. Technical report, 3/07 MIDLAB - Universit di Roma "La Sapienza", 2007.
13. E. Pagani and 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)*, pages 34–42. ACM Press, 1997.
14. S. K. S. Gupta and 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)*, page 111. IEEE Computer Society, 1999.
15. R. Chandra, V. Ramasubramanian, and 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)*, pages 275–283. IEEE Computer Society, 2001.
16. E. Pagani and G. P. Rossi. Providing reliable and fault tolerant broadcast delivery in mobile ad-hoc networks. *Mobile Networks and Applications*, 4(3):175–192, 1999.